KeTop T100

Handterminal

Benutzerhandbuch V 3.1

KEBA®

Automation by innovation.

Hinweise zu diesem Handbuch

Im Handbuch finden Sie an verschiedenen Stellen Hinweise und Warnungen vor möglichen Gefahren. Die verwendeten Symbole haben folgende Bedeutung:

⚠ GEFAHR

- **bedeutet, dass Tod oder schwere Körperverletzung eintreten werden, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.**

⚠ WARNUNG

- **bedeutet, dass Tod oder schwere Körperverletzung eintreten können, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.**

⚠ VORSICHT

- **bedeutet, dass leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.**

VORSICHT

- **bedeutet, dass ein Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.**

Mit dieser Warnung wird auf die möglichen Folgen beim Berühren von elektrostatisch empfindlichen Bauteilen hingewiesen.

Hinweis
Anwendungstipps und nützliche Infos werden mit "Hinweis" gekennzeichnet. Sie enthalten keine Information, die vor einer gefährlichen oder schädlichen Funktion warnt.

© KEBA 2006

Änderungen im Sinne der technischen Weiterentwicklung vorbehalten. Angaben erfolgen ohne Gewähr. Wir wahren unsere Rechte.

Dokument-Version: V 3.1
Dateiname: t100_ge.doc, zuletzt gespeichert am: 25. 10. 2006, Dokument Nr.: 57463

KEBA AG, Postfach 111, Gewerbepark Urfahr, A-4041 Linz
Tel.: ++43 / 732 / 70 90-0, Fax: ++43 / 732 / 73 09 10, E-Mail: keba@keba.com, www.keba.com

KEBA GmbH, Leonhard-Weiss-Straße 40, D-73037 Göppingen
Tel.: ++49 / 7161 / 97 41-0*, Fax: ++49 / 7161 / 97 41-40

KEBA Corp., 100 West Big Beaver Road, Troy, MI 48084
Tel. ++1 / 248 / 526 - 0561, Fax: ++1 / 248 / 526 - 0562, E-Mail: schr@us.keba.com

Änderungsverzeichnis

Ein Beschreibung der Versionsänderungen kann dem Dokument revision.doc entnommen werden.

© KEBA 2006

© KEBA 2006

Inhalt

1 Kurzbeschreibung 7

Zielgruppe dieses Handbuches 7
Allgemeines 7
Bestimmungsgemäßer Gebrauch des Handterminals 7
Aufbau 9
Ergonomie 10
Gehäuse 10
Bedien- und Anzeigefeld 10
Elektronik 11
Typenschild 12

2 Allgemeine Sicherheitsvorschriften 13

3 Allgemeine Hinweise 14

Entsorgungshinweis 14
Handhabung des Handterminals 14

4 Anschluss 16

Anschlussschacht 16
Kabelverlegungen im Anschlussschacht 17
Spannungsversorgung 18
Not-Aus- bzw. grauer Stopp-Schalter 20
Zustimmungseinrichtung 22
Vorhersehbarer Missbrauch des Zustimmungstasters 28
Ethernet 30
RS-422-A 32
Serial Port-Buchse S2 für Debug Schnittstelle (RS-232-C) 34
PC-Card Slot für PC-Cards I, II, III 35

5 Folientastatur 38

Standard 38
LED-Nummerierung 44

6 Display 45

Touch-Screen 45

7 Software 46

Windows CE 46
Programmerstellung für Windows CE 48
KeTop API Design 49
Funktionen 51
Update API Design 64
Initialisierung 65
Programm zum Starten von Applikation und KeTop API 69

© KEBA 2006

RDP – Verbindung via Remote Desktop Protokoll ... 72
KVC – KEBA Virtual Channel ... 80
Remote-Software ActiveSync ... 90
Testtool (VC++ Demo) ... 91

8 Bedienhinweise für das KeTop ... 95

Einstellen von Datum und Uhrzeit ... 95
KeTop Configuration Tool (ConfigTool) ... 96
Überprüfen der Bedien- und Steuerelemente ... 103
Installation von Programmen ... 104
Speichern von Dateien ... 104
Übertragung von Dateien ... 104

9 Optionen ... 107

Override Potentiometer ... 107
Elektronisches Handrad ... 107
Leuchtdrucktaster ... 108
Schlüsselschalter ... 108
Wahlschalter ... 108
Joystick ... 108

10 Zubehör ... 109

Wandhalterung KeTop WB090 und KeTop WB095 ... 109
Wandhalterung mit Höhenverstellung KeTop WB100 / WB110 ... 111
Anschlussbox KeTop CB211 ... 113
Anschlusskabel KeTop TTxxx ... 120
Zwischenkabel KeTop IC2xx ... 121
Downloadkabel KeTop XD040 ... 122

11 Transportbedingungen ... 123

12 Technische Daten ... 124

13 CE Konformität, Richtlinien und Normen ... 127

Richtlinien der Europäischen Union ... 127
Maschinen Sicherheit ... 128
Elektromagnetische Verträglichkeit ... 138
Liste der zutreffenden EG-Richtlinien und angewandten Normen ... 149

KEBA © KEBA 2006

1 Kurzbeschreibung

Zielgruppe dieses Handbuches

Dieses Handbuch richtet sich an die Konstrukteure und Projektierer die dieses Produkt für ihre Maschinen oder Anlagen einsetzen. Für Endbenutzer wie Bedienpersonal hingegen, müssen vom Maschinen und Anlagenhersteller eigene Schulungsunterlagen oder Handbücher erstellt werden.

Allgemeines

Das KeTop Handterminal ist ein tragbares Bedien- und Anzeigegerät im robusten Design mit Windows-CE kompatibler Elektronik.
Durch die Verwendung eines leistungsstarken Prozessors und der Ausstattung mit Ethernet oder serieller Schnittstelle ist das KeTop optimal für verschiedenste Einsätze gerüstet (siehe nächstes Kap. „Bestimmungsgemäßer Gebrauch des Handterminals“).

Mit dem großen Farbdisplay können sämtliche Aufgaben grafisch gelöst werden. Der Touchscreen ermöglicht eine intuitive Bedienerführung.

Anstelle der für raue Betriebsumgebungen nicht geeigneten rotierenden Massenspeichern, wie Disketten- und Festplattenlaufwerke stehen im KeTop skalierbare FLASH- und RAM-Bänke zur Verfügung.

Das KeTop bietet eine Windows CE-Plattform, auf die Applikationen aufgesetzt werden können, die entweder mit gängigen Visualisierungstools oder mit C#, Visual Basic.NET bzw. Visual C++ erstellt worden sind.

Weiters ist es auch möglich, das KeTop als Client an einen Windows NT-, Windows 2000-, oder Windows XP-Server anzubinden.

Durch optionale Bedien- und Steuerelemente kann das KeTop einfach an den jeweiligen Einsatzfall angepasst werden.

Bestimmungsgemäßer Gebrauch des Handterminals

Der bestimmungsgemäße Gebrauch des KeTops erstreckt sich von der Beobachtung und Parametrierung bis hin zur Bedienung von Maschinen, wie zum Beispiel:

- Spritzgießmaschinen
- Roboter
- Werkzeugmaschinen
- Textilmaschinen
- Druckereimaschinen

© KEBA 2006

- Theaterkulissen
- und ähnlichen

in den Normalbetriebsarten, wie beispielsweise

- Automatik

sowie in halbautomatischen oder manuellen Sonderbetriebsarten, wie beispielsweise

- Einrichten
- Teachen
- Testlauf
- und ähnlichen.

Als Sicherheitsfunktionen stehen eine Zustimmungseinrichtung, sowie optional ein Stopp-Schalter oder ein Not-Aus-Schalter zur Verfügung.

Handterminals, die für den temporären Anschluss vorgesehen sind, dürfen keinen rot-gelben Not-Aus-Schalter haben. Für diesen Anwendungsfall steht ein Handterminal mit grauen Stopp-Schalter zur Verfügung.

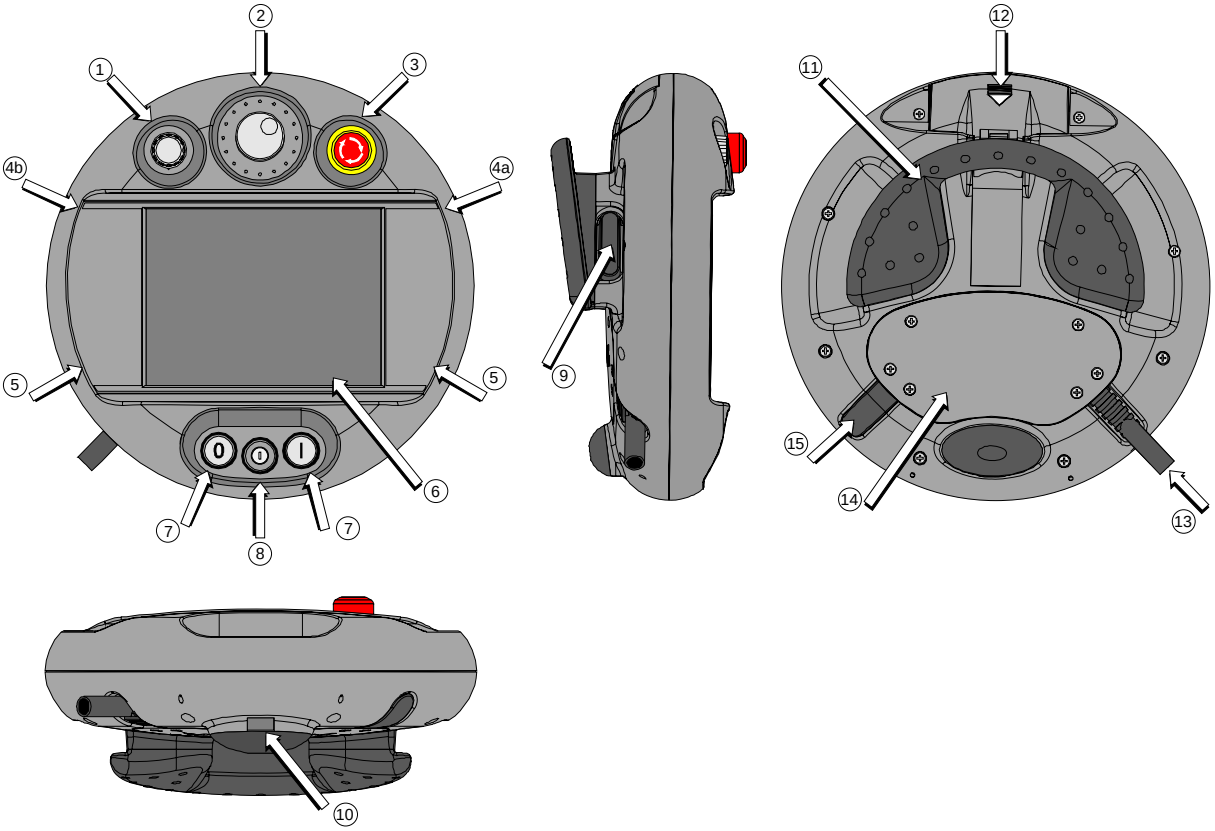
Alle Sicherheitsfunktionen sind zweikreisig ausgeführt, sodass bis zu Sicherheits-Kategorie 3 nach EN 954-1 möglich ist.

Die Auswahl des für die Maschine geeigneten Handterminals sowie die Projektierung der möglichen Zusatzoptionen muss ausgehend von der gesetzlich erforderlichen Gefahren- und Risikobeurteilung im Verantwortungsbereich des Maschinenherstellers erfolgen.

Beachten Sie bitte auch in Bezug auf den bestimmungsgemäßen Gebrauches des Handterminals das Kapitel „CE Konformität, Richtlinien und Normen“.

KEBA © KEBA 2006

Aufbau



1 ..... Override Potentiometer (Option)
2 ..... Elektronisches Handrad oder Joystick (Option)
3 ..... Not-Aus-Taster 2-kreisig oder Stopptaster (Option)
4a ..... 2 Status LEDs (Standard)
4b ..... weitere Status LEDs (Option)
5 ..... Folientastatur mit taktiler Rückmeldung
6 ..... Farbiges Display mit Touch-Screen:
7,7“ STN-LCD VGA-Auflösung (640 x 480) oder
8,4“ TFT mit SVGA-Auflösung (800 x 600)
7 ..... 2 Einbauplätze für (Option):
Leuchtdrucktaster, tastend 0
Leuchtdrucktaster, tastend I
Leuchtdrucktaster, rastend I/0,
8 ..... 1 Einbauplatz für (Option):
Leuchtdrucktaster, tastend 0,
Leuchtdrucktaster, tastend I,
Leuchtdrucktaster, rastend I/0,
Schlüsselschalter mit 3 Stellungen, I-0-II,
Wahlschalter mit 3 Stellungen, I-0-II
9 ..... 2 3-stufige Zustimmungstaster (einer links, einer rechts), jeweils 2-kreisig
10 ..... Funktioneller Multigriff
11 ..... PC-Card-Klappe
12 ..... Zugentlastung und Knickschutz für Anschlusskabel (wird mit Anschlusskabel geliefert)
13 ..... Anschlussschacht
14 ..... Blindstopfen für unbenutzen Kabelauslass (zur Gewährleistung der Schutzart IP54)

HINWEIS: Sämtliche optionalen Bedienelemente sind im Kap. „Optionen“ auf Seite 107 separat beschrieben.

© KEBA 2006

Ergonomie

- Funktioneller Multigriff
- Rundes Gehäuse
- Verschiedene Griffpositionen
- Bedienung für Links- und Rechtshänder
- Bedienung auf Tisch
- Bedienung in Wandhalterung
- Kabelabgangsrichtung links oder rechts von Gehäuse durch einfaches Umlegen selbst bestimmbar
- Gut lesbares Display

Gehäuse

- Vibrations- und schockbeständig
- Gehäuse aus schwer entflammbarem Material (UL 94-V0), schlagfest, beständig gegen Wasser, Reinigungsmittel (Alkohole und Tenside), Öle, Schneideöle (Bohröle), Fette und Schmierstoffe
- Doppelwandiges, extrem robustes Gehäuse. Fallgeprüft aus 1,0 m Höhe auf Industrieboden.

Bedien- und Anzeigefeld

- Folientasten mit mechanischem Druckpunkt
- 2 Status LEDs
- Signalgeber im Gehäuseoberteil
- Resistiver Touch-Screen mit Finger oder mit Stift bedienbar
- Hintergrundbeleuchtetes Farb-LC-Display mit
  7,7“ STN in VGA-Aufkösung (640x480 Pixel) oder
  8,4“ TFT in SVGA-Auflösung (800x600 Pixel)

KEBA © KEBA 2006

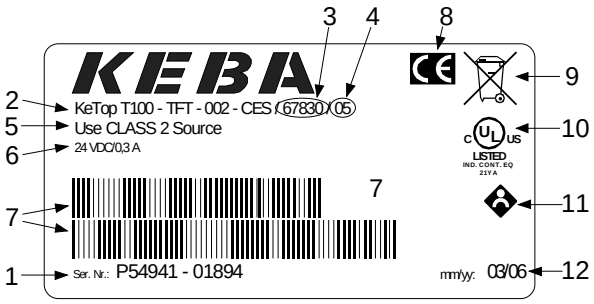
Elektronik

- CPU Intel StrongARM SA-1110/206 MHz
- Speicher-Ausbau:
  - DRAM: max. 128 MB, FLASH: max. 64 MB
- Schnittstellen:
  - Ethernet
  - Serielle Schnittstelle:
    RS-422-A
    RS-232-C (Debugschnittstelle im Gerät)
  - PC-Card-Slot, mit Staubschutzklappe für Gewährleistung von IP54 im geschlossenen Zustand:
    Für den Anschluss verschiedener PC-Card-Komponenten, Typ I-III (DRAM, SRAM, Netzwerkkarten, etc.).

© KEBA 2006

Typenschild

Beschreibung des KEBA-Typenschildes anhand eines Musters.



1 **Seriennummer**
2 **Materialkurzbezeichnung**
3 **Materialnummer**
4 Geräte-Revisions-Nr.
5 Zusatztext (optional)
6 Technische Daten (hier: Spannungsversorgung)
7 Barcode
8 CE-Kennzeichen
9 Hinweis auf Elektronik-Schrott-Verordnung
10 UL-Kennzeichen
11 NSBIV-Kennzeichen
12 Produktionsdatum (Monat/Jahr)

Beschreibung eines KEBA-Typenschildes

© KEBA 2006

2 Allgemeine Sicherheitsvorschriften

Das Handterminal wurde unter Beachtung der ergonomischen Richtlinien sowie den einschlägigen Sicherheitsnormen entwickelt, gefertigt, geprüft und dokumentiert. Bei Beachtung der für den bestimmungsgemäßen Gebrauch beschriebenen Anweisungen und sicherheitstechnischen Hinweise (siehe Kap. „Bestimmungsgemäßer Gebrauch des Handterminals“) gehen deshalb vom Produkt im Normalfall keine Gefahren in Bezug auf Sachschäden oder für die Gesundheit von Personen aus.

Die in diesem Handbuch enthaltenen Anweisungen müssen in jedem Fall genau befolgt werden. Andernfalls können Gefahrenquellen geschaffen oder die im Handterminal integrierten Sicherheitseinrichtungen unwirksam gemacht werden.

Unabhängig von den in diesem Handbuch angeführten Sicherheitshinweisen sind die dem jeweiligen Einsatzfall entsprechenden Arbeitssicherheits- und Unfallverhütungsvorschriften zu beachten.

⚠ WARNUNG

- **Die richtige Projektierung des Handbediengerätes ist durch den Maschinenhersteller aufgrund der Gefahren- und Risikoanalyse durchzuführen. Folgende Sicherheitsaspekte müssen hierfür überlegt werden:**

  - Richtige Kabellänge für Arbeitsbereichseinschränkung
  - Not-Aus oder Stopp-Schalter notwendig bzw. zulässig
  - Sicherheits-Kategorie für die jeweilige Anwendung ausreichend

- **Das Gerät darf nur im einwandfreien Zustand, und unter Beachtung der Betriebsanleitung betrieben werden.**

- **Der Bediener muss dem erforderlichen Ausbildungsniveau genügen, sowie die Einzelheiten der bestimmungsgemäßen Verwendung entsprechend der Bedienungsanleitung kennen.**

- **Die Sicherheitshinweise in den Folgekapiteln sind unbedingt mit zu berücksichtigen**

- **Weitere wichtige Informationen zur Sicherheit und EMV befinden sich im Kapitel „CE Konformität, Richtlinien und Normen“ und sind unbedingt zu beachten.**

© KEBA 2006

3 Allgemeine Hinweise

Entsorgungshinweis

Achten Sie auf die nationalen Bestimmungen zur Entsorgung von elektronischen Bauteilen!

Handhabung des Handterminals

Sie haben sich für ein hochwertiges Handbediengerät entschieden, welches mit modernster, hochempfindlicher Elektronik ausgestattet ist. Um Fehlfunktionen oder Beschädigungen durch unsachgemäße Handhabung zu vermeiden, beachten Sie unbedingt nachfolgende Hinweise beim Betrieb des Gerätes:

VORSICHT

- **Den Anschlussschacht nur bei abgeschalteter Versorgungsspannung öffnen. Andernfalls können Bauteile zerstört werden oder undefinierte Signalzustände auftreten.**
- **Achten Sie darauf, dass niemand über das Kabel stolpern kann und dadurch das Gerät zu Boden fällt.**
- **Achten Sie darauf, dass das Kabel nicht durch Gegenstände gequetscht und dadurch beschädigt wird.**
- **Vermeiden Sie die Kabelführung über scharfe Kanten, wodurch der Kabelmantel aufgescheuert werden kann.**
- **Legen/hängen Sie das Gerät in die dafür vorgesehene Wandhalterung wenn Sie es nicht benützen.**
- **Achten Sie darauf, dass das Gerät nicht auf der Bedienseite abgelegt wird, und dadurch Bedienelemente mechanisch beschädigt werden.**
- **Legen Sie das Gerät niemals auf instabile Oberflächen / Ablagen. Es könnte herunterfallen und dadurch Schaden nehmen.**
- **Stellen Sie das Gerät niemals in die Nähe von Wärmequellen oder direkter Sonneneinstrahlung.**
- **Vermeiden Sie, dass das Gerät mechanischen Erschütterungen, übermäßig viel Staub, Feuchtigkeit oder starken Magnetfeldern ausgesetzt wird.**
- **Reinigen Sie Gehäuse, Bedienfeld und Bedienelemente nicht mit Lösungsmittel, Scheuermittel oder Scheuerschwämmen. Verwenden Sie dazu ein weiches Tuch, das Sie leicht mit Wasser oder einem milden Reinigungsmittel angefeuchtet haben.**
- **Verhindern Sie, dass Fremdkörper oder Flüssigkeiten in das Geräteinnere gelangen. Kontrollieren Sie periodisch die am Gerät vorhandenen Schutzabdeckungen, die Vollständigkeit der Gehäuseverschraubung sowie Beschädigungen am**

KEBA © KEBA 2006

Gehäuse und Kabeldurchführung.

- **Sollte das Gerät trotzdem einen Fehler haben, so senden Sie es bitte mit einer detaillierten Fehlerbeschreibung an Ihren Lieferanten oder an die vereinbarte Serviceniederlassung.**
- **Ist das KeTop mit einem Touch-Screen ausgestatte, bedienen Sie diesen mit einem Finger oder einem Touch-Stift. Der Touch-Screen darf keinesfalls mit spitzen Gegenständen (zB: Schraubendreher,...) bedient werden, da dies zur Zerstörung des Touch-Screens führt.**

Das Handterminal ist bei geöffnetem Anschlussschacht empfindlich gegen elektrostatische Entladung.

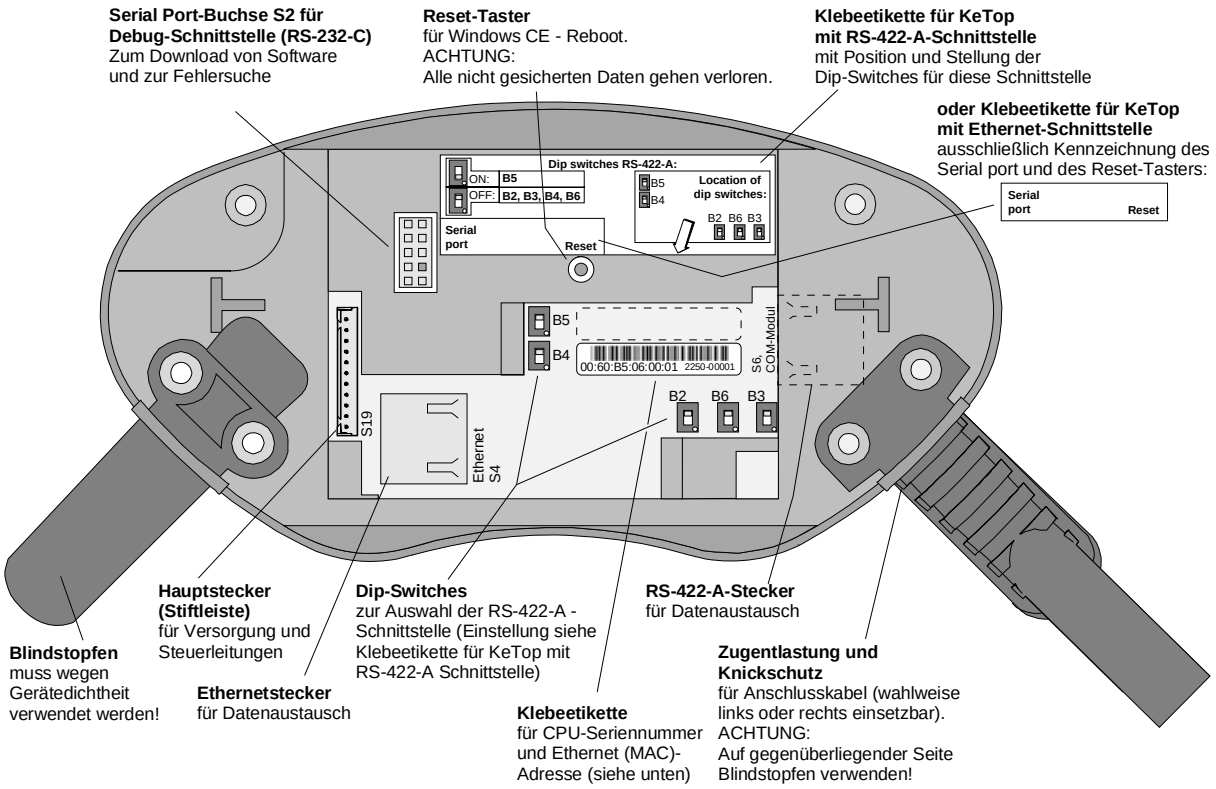
Hinweis

- *Sollte das Gerät zu Boden fallen, so überprüfen Sie, ob die PC-Card-Klappe korrekt geschlossen ist.*
- *Falls im KeTop eine PC-Card verwendet wird, muss im Falle einer heftigen Stoßeinwirkung (zB durch Fallenlassen des KeTops) der korrekte Sitz der PC-Card überprüft werden.*
  (Die PC-Card-Klappe bleibt bei der Stoßeinwirkung zwar geschlossen, die PC-Card selbst kann sich aber aus der Steckposition im Aufnahmeschacht lösen und somit elektrisch nicht mehr richtig kontaktieren.)

© KEBA 2006

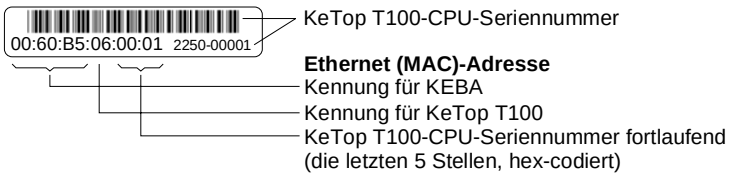
4 Anschluss

Anschlussschacht



Anschlussschacht des KeTop T100

Klebeetikette für CPU-Seriennummer und Ethernet (MAC)-Adresse



Klebeetikette für CPU-Seriennummer und Ethernetadresse

KEBA © KEBA 2006

Kabelverlegungen im Anschlussschacht

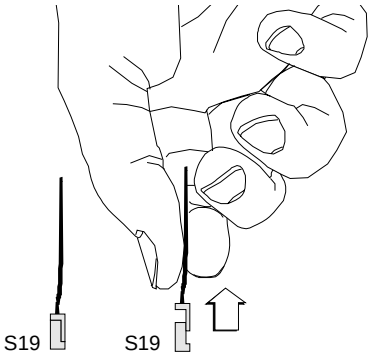
Nach dem Öffnen des Anschlussschachtes können die Anschlussleitungen wie in den nachfolgenden Kapiteln gezeigt, verlegt werden. Bitte beachten Sie vor dem Öffnen des KeTops folgende Hinweise:

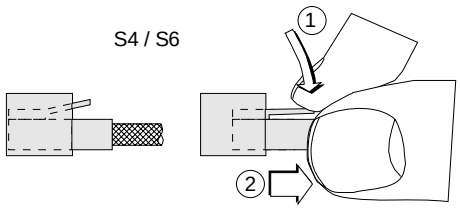
Hinweise für das Öffnen des Anschlussschachtes:

- *KeTop mit dem Display nach unten auf einen planen, sauberen Untergrund auflegen, sodass das KeTop oder dessen Bedienelemente nicht beschädigt werden (zB. ESD-Matte).*
- *Verwenden Sie für das Öffnen und Schließen des Anschlussschachtes einen Schraubendreher der Type „Phillips Kreuzschlitz Größe 2“.*

Hinweise zu Änderungen im Anschlussschacht:

- *Beim Abstecken des Hauptsteckers (S19) ist darauf zu achten, dass der Stecker durch Ziehen mit den Fingern an seinen Adern abgesteckt wird (keine spitzen Gegenstände dafür zu Hilfe nehmen).*
- *Beim Abstecken der RJ-45-Stecker (S4 bzw. S6) ist darauf zu achten, dass der Verriegelungshebel dabei betätigt wird:*





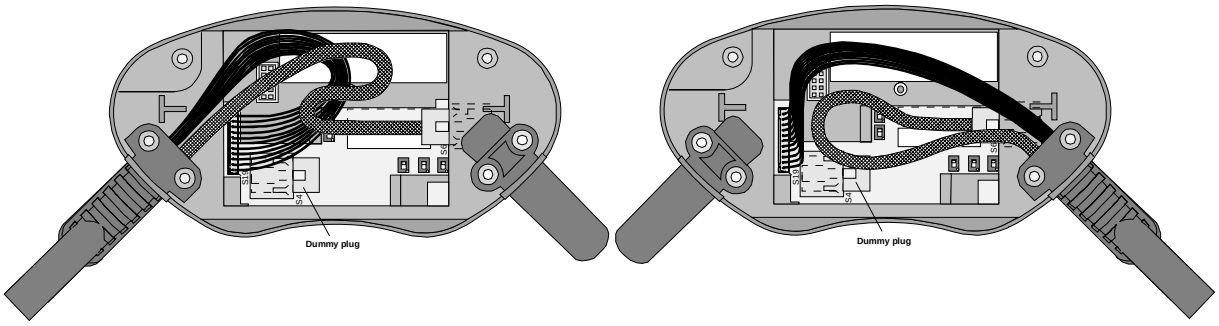
Um Funktionsstörungen zu vermeiden, stellen Sie sicher, dass beim Anstecken der Stecker S19 und S4 / S6 diese korrekt einrasten.

Hinweise zum Schließen des Anschlussschachtes:

- *Die Dichtung muss sauber und unbeschädigt sein und sich an der richtigen Position im Anschlussschachtdeckel befinden*
- *Es dürfen keine Kabel eingeklemmt werden.*
- *Der Anschlussschachtdeckel muss mit allen 6 Schrauben wieder verschraubt werden (Drehmoment: 0,4 bis 0,5 Nm). Nur dadurch kann die entsprechende Schutzart wieder gewährleistet werden.*

© KEBA 2006

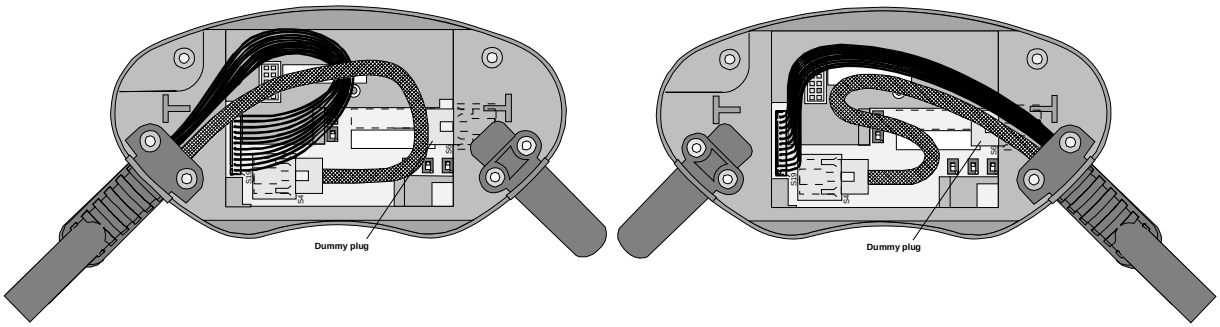
Kabelabgang RS-422-A



Rechts Links

Kabelabgang links und rechts bei Verwendung der RS-422-A-Schnittstelle

Kabelabgang Ethernet



Rechts Links

Kabelabgang links und rechts bei Verwendung der Ethernet-Schnittstelle

Spannungsversorgung

⚠ WARNUNG

- **Das Gerät entspricht der Schutzklasse III nach EN61131-2 bzw. EN50178. Alle Versorgungsspannungen und Schittstellen müssen mit Schutzkleinspannungs-kreisen (nach En 50178 u. EN 61131-2) betrieben werden.**
- **Es muss eine sichere Trennung zwischen Schutzkleinspannung und berührungs-gefährlichen Spannungen gewährleistet werden.**

KEBA © KEBA 2006

VORSICHT

- **Alle Versorgungsstromkreise zum KeTop sind mit max. 3,15 A abzusichern.**

Hinweis

- *Bei der Dimensionierung der Versorgung ist der Spannungsabfall am KeTop TTxxx-Anschlusskabel zu beachten!*

  Spezifikation der Versorgungsleitungen im KeTop TTxxx-Anschlusskabel:

Querschnitt:	*AWG24 (0,24mm²)*
Material:	*verzinkte Kupferlitze*
Leiterwiderstand:	*≤ 90 Ohm/km (≤ 145 Ohm/mile)*

- *Die Versorgungsspannung unmittelbar am Handterminal (ohne KeTop-Anschlusskabel) beträgt nominal:*
  +24 VDC (Versorgungsspannungsbereich: 18-32 VDC).
- *Leistungsaufnahme:*
  7,2 W (400 mA bei 18 V DC, 300 mA bei 24 V DC)
- *Max. Unterbrechungsdauer der Versorgungsspannung:*
  ≤ 10 ms (lt. IEC 61131)

© KEBA 2006

Not-Aus- bzw. grauer Stopp-Schalter

Der Not-Aus- /Stopp-Schalter ist 2-kreisig verdrahtet und die Kontakte sind als Öffner ausgeführt.

Der rot-gelbe Not-Aus am KeTop entspricht den Anforderungen der EN 418. Seine Wirkungsweise muss an Hand der Risikobeurteilung für die Maschine als Stopp der Kategorie 0 oder der Kategorie 1 ausgebildet werden (siehe EN 60204-1 Kapitel 9.2.5.4.2). Die Verschaltung der zwangsöffnenden Schaltkontakte muss jener Sicherheits-Kategorie (nach EN 954-1) genügen, welche an Hand der Risikoanalyse (nach EN 1050) der Maschine festgelegt wird.

Als Option ist das KeTop anstelle mit einem rot-gelben Not-Aus auch mit einem grauen Stopp-Schalter erhältlich. Der graue Stopp-Schalter hat prinzipiell die gleiche Funktionalität, wie der rot-gelbe Not-Aus, und soll durch seine Farbgebung vermeiden, dass bei abgestecktem Handterminal der somit nicht wirkungsvolle Not-Aus bei Gefahr verwendet wird.

Der graue Stopp-Schalter erfüllt ebenfalls alle mechanischen Aspekte der EN 418 und unterscheidet sich nur in der Farbgebung.

⚠ WARNUNG

- **Nicht funktionstüchtige Not-Aus-Einrichtungen können fatale Folgen haben! Rot-gelb gekennzeichnete Not-Aus-Schalter müssen jederzeit und in allen Betriebsarten einer Maschine oder Anlage wirksam sein.**

  Handbediengeräte mit rot-gelbem Not-Aus, welche nicht an einer Maschine angeschlossen sind, müssen so aufbewahrt werden, dass sie nicht sichtbar sind, und somit in einem Not-Fall nicht mit funktionstüchtigen Geräten verwechselt werden können.

  Handbediengeräte, welche zum häufigen temporären An- und Abstecken an Maschinen vorgesehen sind, dürfen daher keinen rot-gelben Not-Aus Schalter aufweisen. Statt dessen ist der graue Stopp-Schalter anzuwenden.

- **Ein Entriegeln der Not-Aus-Einrichtung darf keinen unkontrollierten Wiederanlauf bewirken.**

- **Der Not-Aus ist kein Ersatz für Sicherheitseinrichtungen.**

- **Der Not-Aus am Handbediengerät ist kein Ersatz für die direkt an der Maschine anzubringenden Not-Aus-Schalter.**

- **Bestimmte mechanische Fehler im Not-Aus bzw. Stopp-Schalter können nur bei Betätigung erkannt werden.**

  Nach heftiger Stoßeinwirkung auf das Gerät (zB. durch Fallenlassen), muss der Not-Aus-Schalter auf Funktionsfähigkeit überprüft werden.

  <u>Instandhaltungshinweis:</u>
  Zusätzlich muss der Not-Aus zyklisch (alle 6 Monate) überprüft werden, durch Betätigen des Not-Aus-Schalters und Beobachten, ob die Maschine abschaltet.

KEBA © KEBA 2006

- **Für weitere Informationen zum Not-Aus und Stopp-Schalter ist unbedingt auch das Kapitel „CE Konformität, Richtlinien und Normen“ zu beachten.**

Anschlusswerte

- Anschlussspannung: 24 VDC
- Maximale Stromstärke: 500 mA
- Minimale Stromstärke: 10 mA

© KEBA 2006

Zustimmungseinrichtung

Das KeTop verfügt über zwei Zustimmungstaster, die beidseitig am Gerät angeordnet sind. Dies ermöglicht eine Bedienung sowohl mit der linken als auch mit der rechten Hand. Beide Zustimmungstaster sind parallelgeschaltet und wirken gleichwertig auf die gemeinsamen Sicherheitskreise im Anschlusskabel. Es muss nur ein Taster betätigt werden.

Der Zustimmungstaster besteht aus einem dreistufigen Bedienelement und einer getrennten Auswerteelektronik. Ein wesentliches Merkmal ist die durchgängig zweikreisige Ausführung, beginnend von den Betätigungselementen bis zu den Anschlussklemmen. Die Auswerteschaltungen sind mit unterschiedlichen Technologien und Schaltungen realisiert worden. Durch die elektronische Ausführung der Schaltkontakte ist deren Lebensdauer unabhängig von der Last bis zu deren Nennwerten (ohmsch, induktiv und kapazitiv).

Die Zustimmungstaster - Schaltelemente sind verpolungssicher aufgebaut. Die Ausgänge beider Kreise sind gegen Kurzschluss und Überlast geschützt:

Kreis 1: Thermische Schutzschaltung
Kreis 2: Fold back Kennlinie

Funktionsweise

Das Betätigungselement besteht aus zwei symmetrisch angeordneten Wippen, deren Position durch elektrische Taster ermittelt und an die Auswerteelektronik weitergegeben werden.

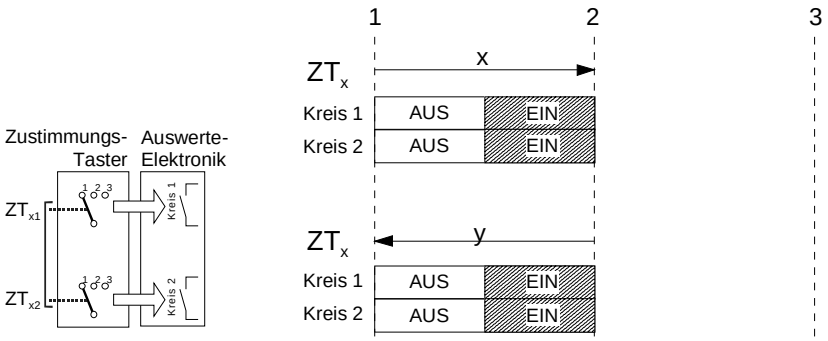
Der Zustimmungstaster kann drei verschiedene Schalterstellungen einnehmen:

Schalterstellung	Funktion	Zustimmtaster	Schaltkontakt
1	Nullstellung	wird nicht betätigt	Aus (geöffnet)
2	Zustimmung	wird betätigt	Ein (geschlossen)
3	Panik	wird durchgedrückt	Aus (geöffnet)

KEBA © KEBA 2006

Normale Betätigung

Nullstellung x Zustimmung y Nullstellung



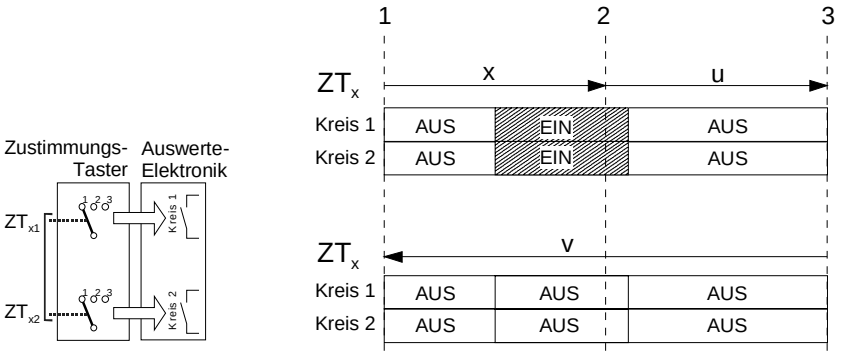
x ... L (Links) oder R (Rechts)

Schaltwegdiagramm für normale Betätigung

Panik-Betätigung

Ein Durchdrücken der Betätigungselemente auf Panik-Stellung wird so ausgewertet, dass beim Loslassen die Zustimm-Stellung übersprungen wird:

Nullstellung x Zustimmung u Panik v Nullstellung



Schaltwegdiagramm für Panik-Betätigung

Hinweis

- *Zustimmungstaster sind beim KeTop immer 2-kreisig ausgeführt.*

Das Erreichen der Sicherheits-Kategorie 3 nach EN 954-1:1996 ist durch die Realisierung der Zustimmungseinrichtung mit 2 Kreisen möglich.

Die Sicherheits-Kategorie 3 bedeutet, dass 1 Fehler nicht zum Sicherheitsverlust führen darf und wann immer in angemessener Weise durchführbar, der einzelne Fehler erkannt wird.

© KEBA 2006

Die Anschlussbeispiele im Kapitel "Anschlussbeispiel mit Überwachungsgerät PILZ PST1" und "Anschlussbeispiel mit Überwachungsgerät ELAN SRB-NA-R-C.27/S1" zeigen, wie mit dem KeTop und dessen sicherheitsbezogenen Teilen die **Sicherheits-Kategorie 3** erreicht werden kann. Es ist zu beachten, dass das gesamte Konzept der Maschine dafür ausgelegt werden muss.

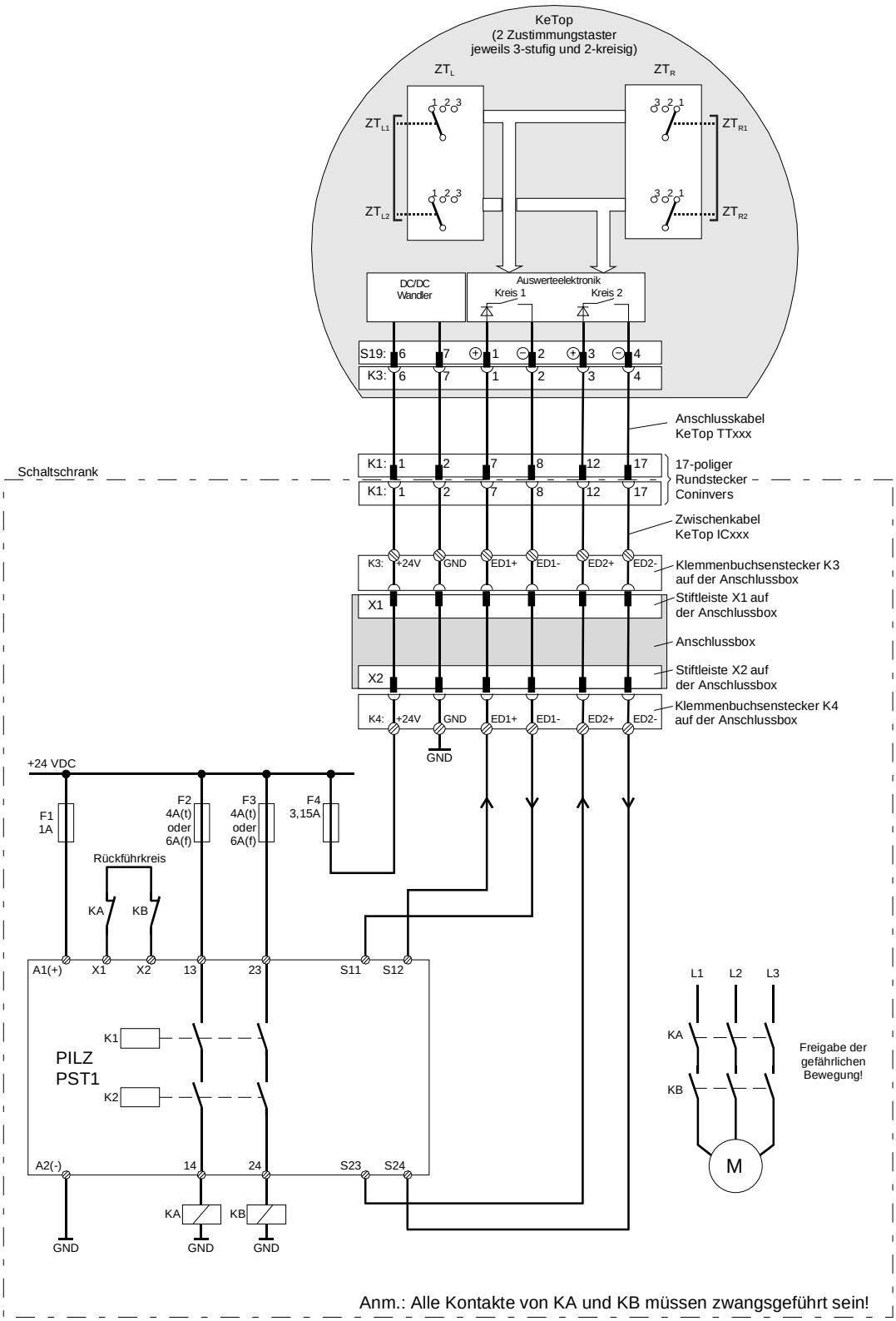
Die Gleichzeitigkeitsüberwachung durch das Überwachungsgerät ist erforderlich, weil es sonst zu einer unerkannten Fehleranhäufung kommen könnte, welche in Folge zum Sicherheitsverlust führen würde:

Beispiel: Wenn ein Kanal der Zustimmungseinrichtung durch einen Fehler auf Zustimmung geht und der zweite Kanal nach unbestimmter Zeit ebenfalls durch einen Fehler Zustimmung gibt, wäre keine Abschaltung durch den Zustimmtaster mehr möglich.

Die EN 60204-1:1997 schreibt weiter vor, dass die Zustimmungseinrichtung an einen Stopp der Kategorie 0 oder 1 anzuschließen ist, d.h. dass die Energie abgeschaltet werden muss.

KEBA © KEBA 2006

Anschlussbeispiel mit Überwachungsgerät PILZ PST1



Schaltungsvorschlag: Verschaltung der Zustimmungstaster für Sicherheits-Kategorie 3 mit PILZ-Überwachungsgerät. Es ist die Betriebsanleitung der Fa. PILZ zum Gerät PST1 zusätzlich zu beachten.

© KEBA 2006

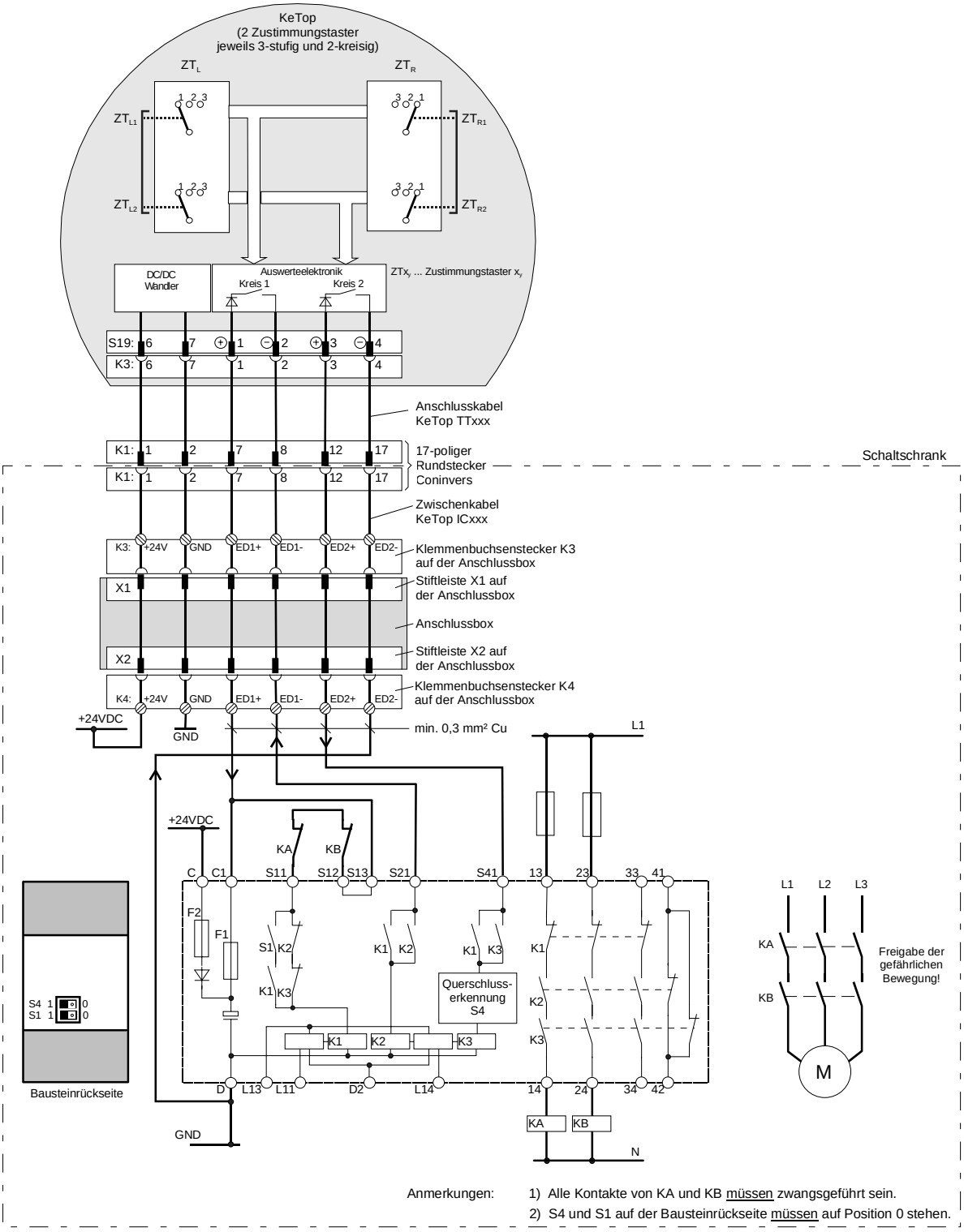
Funktionsablauf:

- Nur wenn beide Zustimmkreise „gleichzeitig“ geschlossen werden (durch Betätigung eines der Zustimmungstaster), gehen die beiden Ausgangsrelais K1 und K2 in Arbeitsstellung und die Ausgangskontakte 13-14 und 23-24 schließen.
- Die Ausgangsrelais K1 und K2 gehen nicht in Arbeitsstellung, wenn
  - aufgrund eines Fehlers nur ein Zustimmungskreis geschlossen wird
  - der Toleranzwert für die Gleichzeitigkeit überschritten wird
  - der Rückführkreis X1-X2 offen ist
- Wird nach der Aktivierung ein Zustimmungskreis unterbrochen (durch Loslassen des Zustimmungstasters oder durch einen Fehler), gehen die Ausgangsrelais K1 und K2 wieder in die Ruhestellung.
  Die zwangsgeführten Kontakte zwischen den Ausgangsklemmen 13-14 und 23-24 öffnen. Die Ausgangsrelais sprechen erst wieder an, wenn beide Zustimmungskreise geöffnet und erneut gleichzeitig geschlossen werden.

Es wird somit die Anforderung erfüllt, dass ein einzelner Fehler nicht zum Verlust der Sicherheitsfunktion führt. Ein einzelner Fehler wird spätestens beim nächsten Zyklus erkannt.

KEBA © KEBA 2006

Anschlussbeispiel mit Überwachungsgerät ELAN SRB-NA-R-C.27/S1



Schaltungsvorschlag: Verschaltung der Zustimmungstaster für Sicherheits-Kategorie 3 mit ELAN-Überwachungsgerät SRB-NA-R-C.27/S1. Es ist die Betriebsanleitung zum Gerät SRB-NA-R-C.27/S1 zusätzlich zu beachten.

© KEBA 2006

Funktionsablauf:

- Nur wenn beide Zustimmkreise „gleichzeitig“ geschlossen werden (durch Betätigung eines der Zustimmungstaster), gehen die beiden Ausgangsrelais K2 und K3 in Arbeitsstellung und die Ausgangskontakte 13-14, 23-24 und 33-34 schließen.
- Die Ausgangsrelais K2 und K3 ziehen nicht an, wenn
  - aufgrund eines Fehlers nur ein Zustimmungskreis geschlossen wird
  - der Toleranzwert für die Gleichzeitigkeit überschritten wird
  - der Rückführkreis S11-S12 offen ist
- Wird nach der Aktivierung ein Zustimmungskreis unterbrochen (durch Loslassen des Zustimmungstasters oder durch einen Fehler), gehen die Ausgangsrelais K2 und K3 wieder in die Ruhestellung. Die zwangsgeführten Kontakte zwischen den Ausgangsklemmen 13-14 und 23-24 öffnen. Die Ausgangsrelais sprechen erst wieder an, wenn beide Zustimmungskreise geöffnet und erneut gleichzeitig geschlossen werden.

Es wird somit die Anforderung erfüllt, dass ein einzelner Fehler nicht zum Verlust der Sicherheitsfunktion führt. Ein einzelner Fehler wird spätestens beim nächsten Zyklus erkannt.

Vorhersehbarer Missbrauch des Zustimmungstasters

Als vorhersehbarer Missbrauch versteht man das unerlaubte Fixieren des Zustimmungstasters in der Zustimmstellung mit Hilfsmitteln. Dieser vorhersehbarer Missbrauch ist einzuschränken. Folgende Maßnahmen, die den Stillstand der Maschine im Handbetrieb zur Folge haben, werden dazu empfohlen:

- Abfrage des Zustimmungstasters beim Einschalten der Maschine/Anlage und Abfrage des Zustimmungstasters beim Wechsel der Betriebsart von Automatik auf Manuell (Handbetrieb). (Zustimmtaster darf nicht in Zustimmstellung sein.)
- Der Zustimmungstaster muss innerhalb eines festgelegten Zeitraumes losgelassen und erneut in Zustimmstellung gebracht werden. Die Länge des Zeitraumes ist je nach Tätigkeitsanforderung zu wählen.

KEBA © KEBA 2006

Technische Daten der Zustimmungstaster-Schaltelemente

Nennspannung	24 V DC (typ.)
	32 V DC (max.)
Nennstrom	500 mA (typ.)
Kurzschlussstrom	Kreis 1: max. 1,9 A
	Kreis 2: max. 600 mA
max. induktive Last (bei 500 mA)	Kreis 1: max. 1H
	Kreis 2: max. 320 mH
max. kapaz. Last	Kreis 1: keine Grenze, da Transistor thermisch geschützt ist
	Kreis 2: max. 500 µF

Die Zustimmungstaster - Schaltelemente sind verpolungssicher aufgebaut.
Die Ausgänge beider Kreise sind gegen Kurzschluss und Überlast geschützt:

Kreis 1: Thermische Schutzschaltung
Kreis 2: Fold back Kennlinie

⚠ WARNUNG

- **Der Zustimmungstaster ist als Schutzfunktion nur dann geeignet, wenn die den Zustimmungstaster betätigende Person eine Personengefährdung rechtzeitig erkennt und dann sofort Maßnahmen zur Vermeidung von Gefahren treffen kann! Als Zusatzmaßnahme kann reduzierte Geschwindigkeit der Bewegung erforderlich sein. Die zulässige Geschwindigkeit muss an Hand einer Risikobeurteilung ermittelt werden.**
- **Mit einem Zustimmungstaster alleine dürfen keine Befehle für gefahrbringende Zustände eingeleitet werden. Hierzu ist ein zweiter bewusster Startbefehl erforderlich (Taste am Handbediengerät).**
- **Es darf sich nur jene Person im Gefahrenbereich aufhalten, die den Zustimmungstaster betätigt.**
- **Für weitere Informationen zur Zustimmungseinrichtung ist unbedingt auch das Kapitel „CE Konformität, Richtlinien und Normen“ zu beachten.**

© KEBA 2006

Ethernet

Die Ethernet-Schnittstelle ist standardmäßig im KeTop vorhanden, basiert auf der 10BaseT Spezifikation und ist für Halbduplex-Betrieb geeignet.

Die Datenkommunikation für diese Schnittstellen erfolgt über den Ethernet-Stecker S4 im Anschlussschacht des KeTops.

Folgende Schnittstellenparameter sind fest vorgegeben:

- 10 Mbaud
- Protokoll TCP/IP

Die Ethernet – Schnittstelle wird unter Windows CE wie folgt ausgewählt:

Start -> Settings -> Control Panel -> Network and Dial-up Connections:
Hier kann die Ethernet-Schnittstelle *SMSC91C9X* ausgewählt werden.

⚠ WARNUNG

- **Wenn das KeTop und die Steuerung nicht mit einer Punkt-zu-Punkt-Verbindung kommunizieren, kann es passieren, dass es bei der Übertragung von zum Beispiel Tastaturdaten zu einer Verzögerung kommen kann.**

  Es wird in diesem Fall daher dringend empfohlen, die Verbindung von Steuerung und KeTop nur über einen Ethernet-Switch herzustellen, was eine Punkt-zu-Punkt-Verbindung ermöglicht.

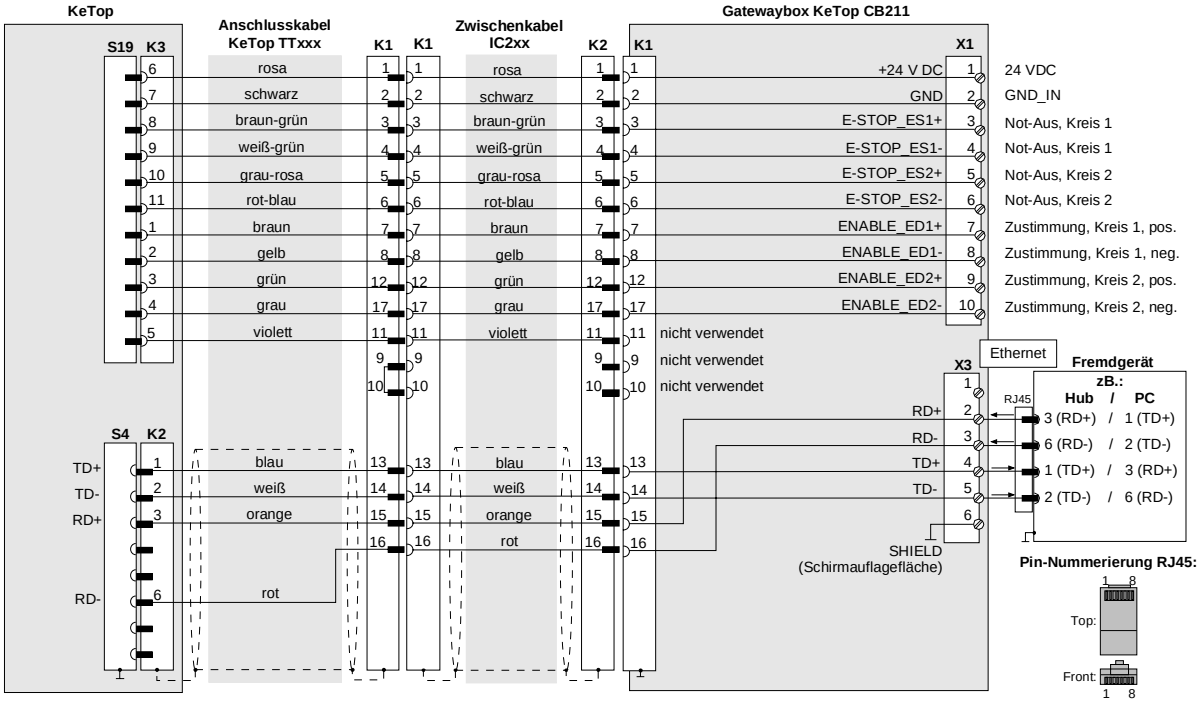
Wird in den PC-Card-Slot eine Ethernet-PC-Card eingeschoben, dann muss die interne Ethernet-Schnittstelle deaktiviert werden.

Die Auswahl zwischen interner Ethernet- oder PC-Card-Ethernet-Schnittstelle erfolgt im „ConfigTool“. Siehe dazu Kap. „KeTop Configuration Tool (ConfigTool)“ auf Seite 96).

Die Stellungen der Dip-Schalter im Anschlussschacht sind für diese Schnittstelle nicht relevant.

KEBA © KEBA 2006

Anschlussplan



Ethernet-Anschluss des KeTop über Anschlussbox KeTop CB211

© KEBA 2006

RS-422-A

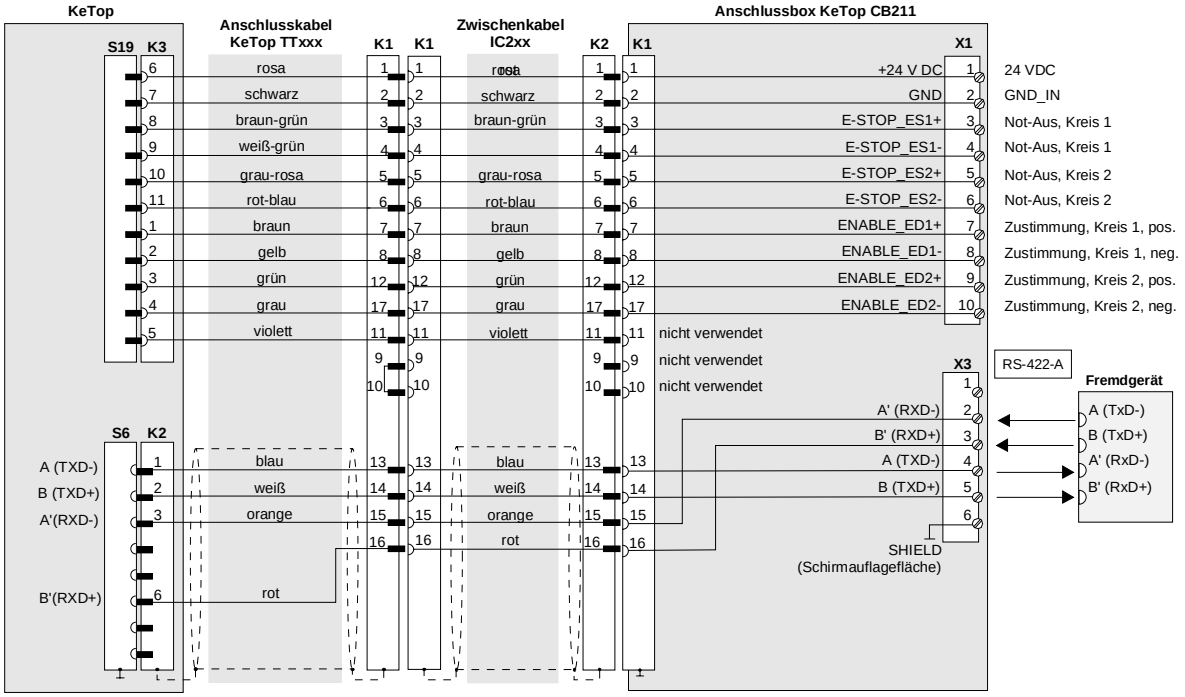
Die serielle Datenkommunikation für diese Schnittstelle erfordert werksseitig den Einbau eines optionalen Zusatzprintes im KeTop.
Die Kommunikation erfolgt über den COM-Modul-Stecker S6 im Anschlussschacht des KeTops.

Hinweis

- *Bei Verwendung der RS-422-A Schnittstelle ist keine gleichzeitige Verwendung der Ethernet-Schnittstelle möglich.*

Die RS-422-A-Schnittstelle ist in der Software dem Schnittstellenport COM 5 zugewiesen. Die Einstellung der Schnittstellenparameter erfolgt über das WIN32API im Windows-Betriebssystem.

Anschlussplan



Dip switch im KeTop T100 für RS-422-A-Betrieb:

B2	B3	B4	B5	B6
OFF	OFF	OFF	ON	OFF

RS-422-A-Anschluss des KeTop über Anschlussbox KeTop CB211

© KEBA 2006

Allgemeine Informationen zur RS-422-A - Schnittstelle

- Der Spannungspegel der Leitung A des Senders muss bei einer binären 1 (Ruhezustand der Schnittstelle) negativ gegenüber Leitung B sein.
- Der Spannungspegel der Leitung A des Senders muss bei einer binären 0 (Aktivzustand der Schnittstelle) positiv gegenüber Leitung B sein.

Zur Identifizierung der Leitungen, kann mittels eines Voltmeters die Spannung zwischen den Leitungen A und B gemessen werden.

© KEBA 2006

Serial Port-Buchse S2 für Debug Schnittstelle (RS-232-C)

Diese serielle Schnittstelle kann zum Debuggen oder Download von Software per Boot-Loader verwendet werden, oder zum Abgleichen bzw. Übertragen von Daten zu einem PC mittels der Remote-Software ActiveSync von Microsoft. Hierfür ist das als Zubehör erhältliche serielle Downloadkabel KeTop XD040 erforderlich.

Folgende Schnittstellenparameter sind fest vorgegeben:

- 38400 Baud
- 8 Datenbits
- 1 Stoppbit
- no parity
- no handshake

Die Debug-Schnittstelle ist in der Software dem COM 1-Schnittstellenport zugewiesen.

Die Stellungen der Dip-Schalter im Anschlussschacht sind für diese Schnittstelle nicht relevant.

KEBA © KEBA 2006

PC-Card Slot für PC-Cards I, II, III

Die folgende Auflistung gibt eine Übersicht über die von KEBA getesteten PC-Cards und deren Herstelleradressen. Folgende PC-Cards können daher für den Einsatz im KeTop empfohlen werden:

ATA-Flashkarten:

- Hersteller: Kingston, SanDisk, Kingmax, Viking
  Type: alle PC-Card Typ II
  Speichergrößen: ab 8 MB

PC-Card Adapter für CompactFlash (CF) Karten:

- Hersteller: Hama
  Type: PC-Card Adapter CF Type II, Hama Art. Nr. 56949

- Hersteller: Kingston
  Type: CF/ADP Compact Flash PC Card Adapter

- Hersteller: Ultron
  Type: UPA-150 PCMCIA Adapter für Compact Flash Card

Ethernetkarten:

- Hersteller: Socket (http://www.socketcom.com)
  Bezeichnung: LP-E Ethernet Card
  Type: PC-Card Typ II
  Übertragungsrate: 10 Mbit/s
  Besonderheiten: NE2000 komatibel

- Hersteller: Orinoco (http://www.wavelan.com)
  Bezeichnung: WaveLan Silber IEEE 802.11
  Übertragungsrate: 11 Mbit/s
  Besonderheiten: Wireless LAN-Card

Modemkarten:

- Hersteller: TDK
  http://www.tdksys.com/Products/5660.html
  Bezeichnung: Global Freedom 5660
  Type: PC-Card Typ II

Hinweis

- *CardBus-Cards werden vom KeTop nicht unterstützt.*
- *PC-Cards, die 12 V Programmierspannung benötigen, werden nicht unterstützt.*
- *SRAM-Karten werden von Windows CE nicht unterstützt.*

© KEBA 2006

Einsetzen der PC-Card

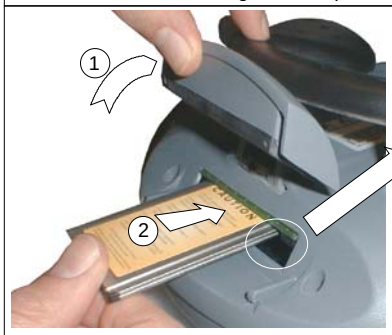
1

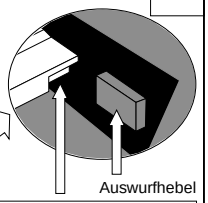
KeTop mit dem Display nach unten auf einem planen, sauberen Untergrund auflegen, sodass das KeTop oder dessen Bedienelemente nicht beschädigt werden (zB.: ESD-Matte)

2

Verriegelung der PC-Card-Klappe wie abgebildet entriegeln (es muss dabei der Druckpunkt überwunden werden).

3





ACHTUNG:
Beim Einsetzen darauf achten, dass der Anschlag der PC-Card auf der Seite des Auswurfhebels in den Aufnahmeschacht eingeführt wird.

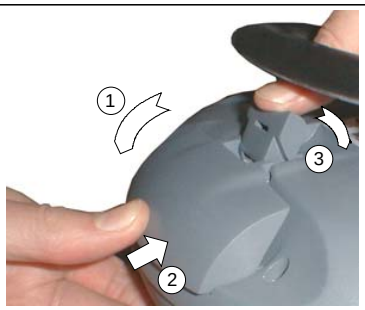
1) Öffnen der Klappe
2) Einsetzen der PC-Card wie dargestellt

4

ACHTUNG:
Überprüfen Sie vor dem Schließen der PC-Card-Klappe den ordnungsgemäßen Zustand und den korrekten Sitz der Dichtung.

PC-Card soweit in den Schacht einführen bis die Verriegelung einrastet und der Entriegelungsknopf ausgefahren ist.

5



1) Schließen der Klappe
2) und 3) Verriegeln der Klappe wie dargestellt

6

Verriegelung einrasten!

Verriegelung der Klappe ganz nach unten drücken, bis sie fühlbar einrastet. Nur dann ist die Schutzart IP54 wieder gewährleistet.

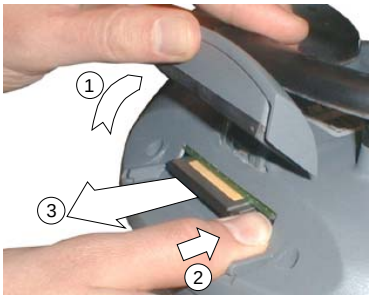
© KEBA 2006

Entnahme der PC-Card

1

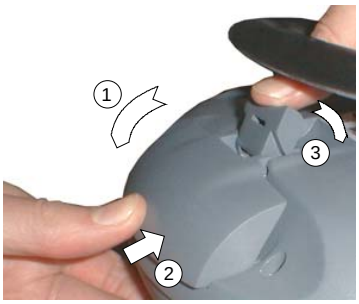
Verriegelung der PC-Card-Klappe wie abgebildet entriegeln (es muss dabei der Druckpunkt überwunden werden).

2



1) PC-Card-Klappe öffnen
2) Auswurfmechanismus der PC-Card-Aufnahme betätigen
3) PC-Card entnehmen

3



1) Schließen der Klappe
2) und 3) Verriegeln der Klappe wie dargestellt

4

Verriegelung der Klappe ganz nach unten drücken, bis sie fühlbar einrastet. Nur dann ist die Schutzart IP54 wieder gewährleistet.

© KEBA 2006

5 Folientastatur

Standard

Allgemeine Folientastatur

Tastaturbelegung

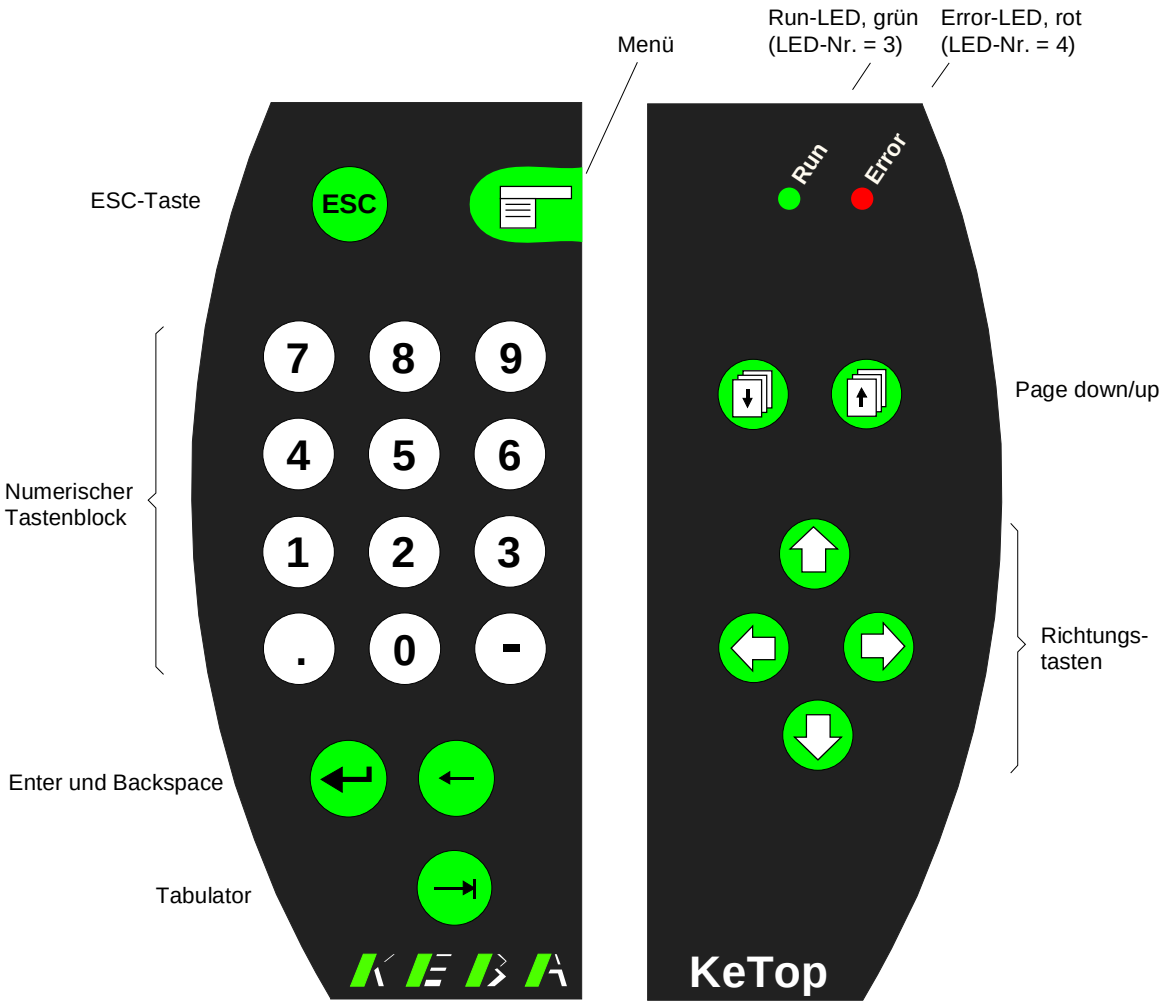
Die Tastaturbelegung ist in der Registry gespeichert und ist für die Allgemeine Folientastatur wie folgt festgelegt:

Tastenbeschriftung	Windows Virtual Key Code	Hex	Dez
ESC	VK_ESCAPE	0x1B	27
Menü	VK_MENU	0x12	18
Numpad Zahlen 0-9	VK_0-9	0x30 - 0x39	48 - 57
Punkt (.)	VK_PERIOD	0xBE	190
Minus (-)	VK_OEM_MINUS	0xBD	189
Enter	VK_RETURN	0x0D	13
Backspace	VK_BACK	0x08	8
Tabulator	VK_TAB	0x09	9
Page Down	VK_NEXT	0x22	34
Page Up	VK_PRIOR	0x21	33
Cursor Up	VK_UP	0x26	38
Cursor Links	VK_LEFT	0x25	37
Cursor Rechts	VK_RIGHT	0x27	39
Cursor Down	VK_DOWN	0x28	40
Linker Leuchtdrucktaster	VK_F13	0x7C	124
Schlüsselschalter/Wahlschalter Stellung links	VK_F14	0x7D	125
Schlüsselschalter/Wahlschalter Stellung rechts	VK_F15	0x7E	126
Rechter Leuchtdrucktaster	VK_F16	0x7F	127

Je nach Displaygröße des KeTops wird jeweils zwischen 2 unterschiedlich großen Folientastaturen unterschieden:

KEBA © KEBA 2006

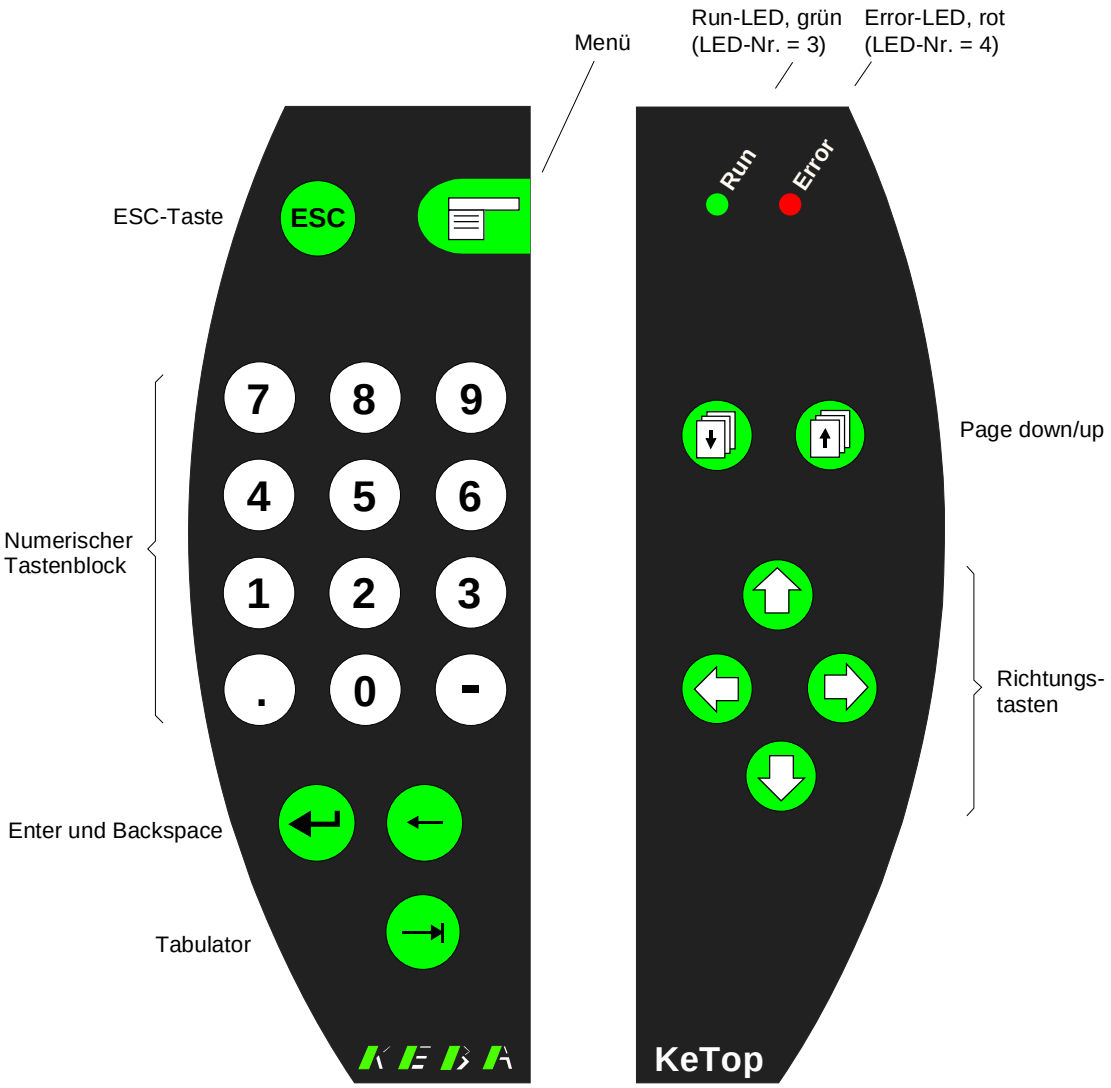
a) Displaygröße 7,7“ (640x480 Pixel)



Allgemeine Folientastatur für 7,7“ Display

© KEBA 2006

b) Displaygröße 8,4“ (800x600 Pixel)



Allgemeine Folientastatur für 8,4“ Display

© KEBA 2006

Robotik-Folientastatur

Tastaturbelegung

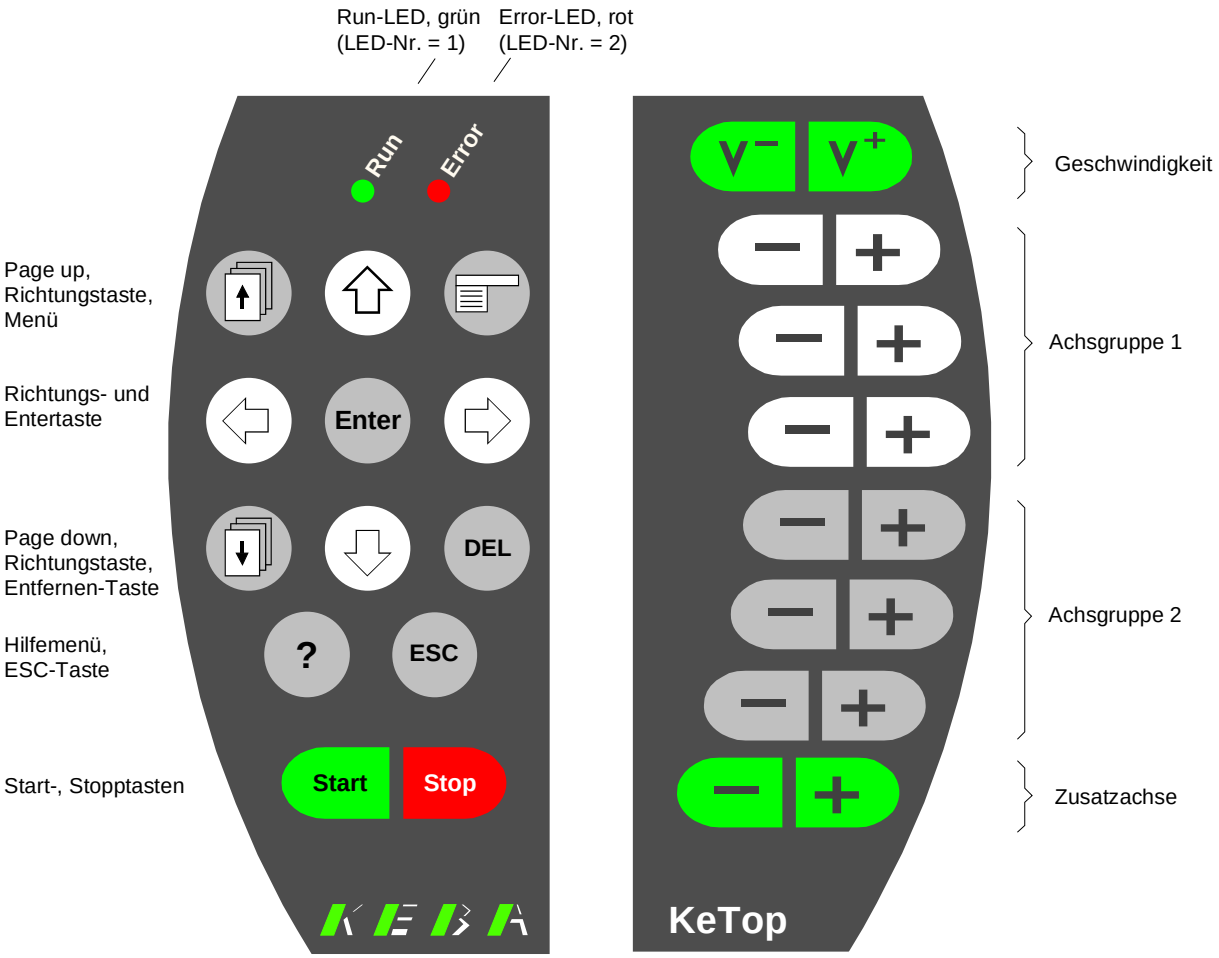
Die Tastaturbelegung ist in der Registry gespeichert und ist für die Robotik-Folientastatur wie folgt festgelegt:

Tastenbeschriftung	Windows VK Code	Hex	Dez
Page Down	VK_NEXT	0x22	34
Page Up	VK_PRIOR	0x21	33
Cursor Up	VK_UP	0x26	38
Cursor Links	VK_LEFT	0x25	37
Cursor Rechts	VK_RIGHT	0x27	39
Cursor Down	VK_DOWN	0x28	40
Menü	VK_MENU	0x12	18
Enter	VK_RETURN	0x0D	13
Entfernen	VK_DELETE	0x2E	46
ESC	VK_ESCAPE	0x1B	27
Hilfe (?)	VK_SHIFT + VK_SLASH	0x10 + 0xBF	16 + 191
Start	VK_1	0x31	49
Stop	VK_0	0x30	48
V-	no VK Code defined	0xC1	193
V+	no VK Code defined	0xC2	194
A1-	no VK Code defined	0xC3	195
A1+	no VK Code defined	0xC4	196
A2-	no VK Code defined	0xC5	197
A2+	no VK Code defined	0xC6	198
A3-	no VK Code defined	0xC7	199
A3+	no VK Code defined	0xC8	200
A4-	no VK Code defined	0xC9	201
A4+	no VK Code defined	0xCA	202
A5-	no VK Code defined	0xCB	203
A5+	no VK Code defined	0xCC	204
A6-	no VK Code defined	0xCD	205
A6+	no VK Code defined	0xCE	206
Z+	no VK Code defined	0xCF	207
Z-	no VK Code defined	0xD0	208

Je nach Displaygröße des KeTops wird jeweils zwischen 2 unterschiedlich großen Robotiktastaturen unterschieden:

© KEBA 2006

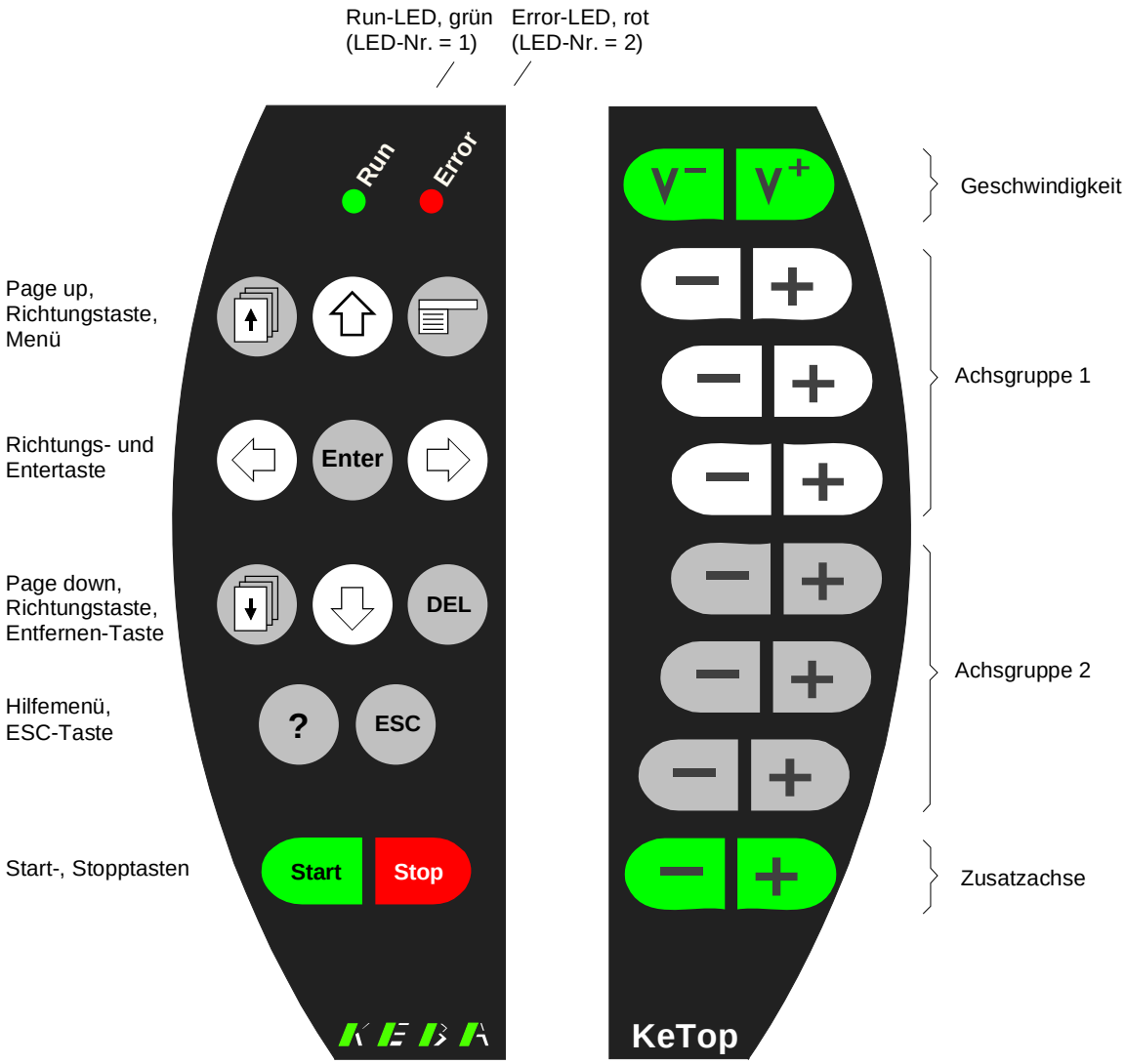
a) Displaygröße 7,7“ (640x480 Pixel)



Robotik Folientastatur für 7,7“ Display

© KEBA 2006

b) Displaygröße 8,4“ (800x600 Pixel)



Robotik Folientastatur für 8,4“ Display

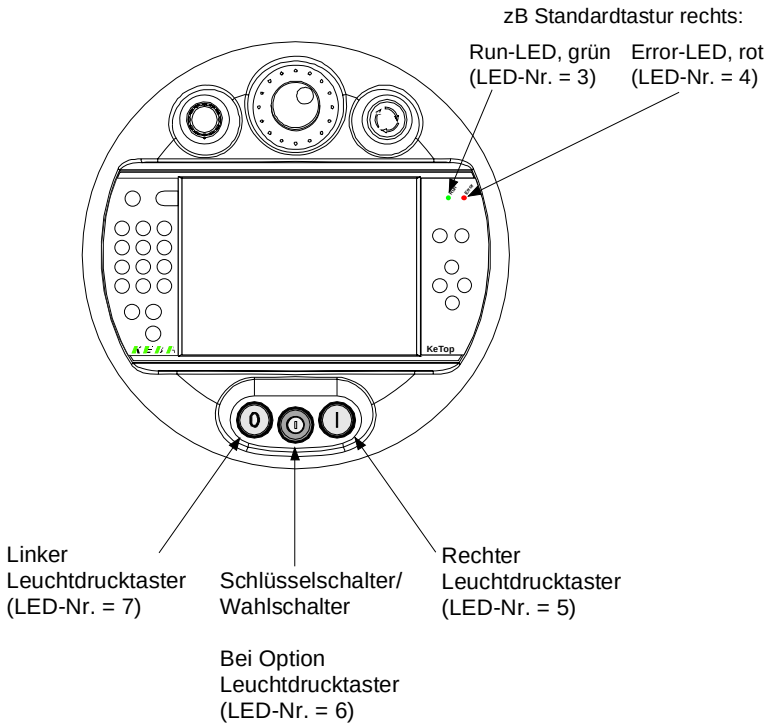
© KEBA 2006

LED-Nummerierung

Aus diesem Kapitel ist die LED-Nummerierung der Tastatur-LEDs und der Leuchtdrucktaster-LEDs ersichtlich. Die LED-Nummer wird zur Ansteuerung der LED benötigt.

Die Ansteuerung einer LED kann auf 2 Arten erfolgen:

- mittels *KeTop API design*
  (vgl. Kapitel „**LED-Funktionen** -> **KtpSetKeyboardLed**“ auf Seite 57)
- oder mittels *KVC – KEBA Virtual Channel*
  (vgl. Kapitel „Beispiel: Interface am Server“ auf Seite 84)



LED-Nummerierung am KeTop

© KEBA 2006

6 Display

Das KeTop ist mit zwei in Auflösung und Bildschirmdiagonalen unterschiedlichen Displays erhältlich:

	Farbdisplay 7,7“	Farbdisplay 8,4“
Typ:	grafikfähiges Farb-STN-LCD	grafikfähiges Farb-TFT-LCD
Größe:	7,7“ (160 x 120 mm)	8,4“ (175 x 130 mm)
Auflösung:	VGA 640 x 480 Pixel	SVGA 800 x 600 Pixel
Darstellung:	256 Farben	65535 Farben
Hintergrundbeleuchtung:	CCFT Kaltkathodenröhre	CCFT Kaltkathodenröhre
Touch-Screen:	analog-resistiv	analog-resistiv

Die Lebensdauer der Hintergrundbeleuchtung (40.000 h) kann über die Systemsteuerung mittels Aktivierung eines Bildschirmschoners deutlich verlängert werden. Gehen Sie dazu wie folgt vor:

Start -> Settings -> Control Panel -> Display -> Folder Backlight
Auswahl der Funktion „Automatically turn off backlight while on external power“

Hier kann die Eingabe der Abschaltzeit erfolgen (15s bis 30min).

Touch-Screen

Wenn der Touch-Screen nicht mit dem Finger bedient wird, ist die Verwendung eines Touch-Stiftes empfohlen. Es eignen sich hierfür handelsübliche Touchstifte von PDAs wie zum Beispiel von PALM und SONY.

Im Auslieferungszustand ist der Touch-Screen des KeTops vorkalibriert, sodass eine Kalibrierung nicht mehr notwendig ist.
Sollte sich aus irgendwelchen Gründen (Luftfeuchtigkeit, Temperatur,...) die Notwendigkeit ergeben, eine Kalibrierung durchzuführen, führen Sie unter Windows CE einen Kalibriervorgang durch:

Start -> Settings -> Control Panel -> Stylus -> Calibration -> Recalibrate

Der Kalibriervorgang kann auch über das ConifgTool durchgeführt werden.
Siehe dazu Kap. „Display- und Touchscreeneinstellungen“ auf Seite 98.

© KEBA 2006

7 Software

Windows CE

Das KeTop wird mit einer vorinstallierten Software ausgeliefert, welche fix im Flash des Gerätes gespeichert ist. Diese Software wird auch Image genannt, da sie alle Softwareinhalte in einem einzigen File vereint. Das Image enthält den Windows CE-Betriebssystemkern und eventuell zusätzliche Anwendungen. Es wird mit dem MS Platformbuilder für Windows CE erstellt.
Diese Beschreibung gilt für Images ab Version 2.10.
(Versionsanzeige am KeTop über: Start -> Programs -> KeTop -> Version)

Das KeTop ist standardmäßig mit folgender Software ausgestattet

Betriebssystem Windows CE 4.2:

- Systemsteuerung
- Command Prompt
- Windows Explorer
- Pocket Internet Explorer

Flash File System:

- IPSM (Intel Persistent Storage Manager)

Vorinstallierte Anwendungsprogramme:

- Pocket Notepad
- RDP-Client (Remote Desktop Connection)
- Pocket Registry Editor (\windows\regedit.exe)
- Active Sync

Vorinstallierte Treiber:

- PC-Card Wireless LAN-Karte Type Orinoco PC Card Silver 11Mbit/s IEEE 802.11b
- PC-Card Modemkarte Windows CE Standard

KeTop spezifische SW: (unter: Start -> Programs -> KeTop)

- KeTop Configuration Tool
- KeTop Virtual Channel (KVC)
- Registry Backup
- Set Time, temporäre Zeit- /Datumseingabe
- System Check, Testtool für KeTop spezifische Funktionen

KEBA © KEBA 2006

- TouchClean, Touch Cleaning Application
- Version, Versionsübersicht
- Toggle-Sip (Software-Keyboard)
- KeTop Tools (Erweiterung in Systemsteuerung für Konfiguration von Kontrast, Helligkeit, Backlight, Bildschirmschoner, Touch- und Joystick-kalibrierung, Löschen der Registry, Image-Update, ..)
- System Check, Testtool für KeTop spezifische Funktionen

Test Tools

- TestSerial, Testprogramm für serielle Schnittstellen (\windows\testserial.exe)
- Reset (\windows\reset.exe)

© KEBA 2006

Programmerstellung für Windows CE

Der Anwender kann sehr einfach eigene Programme für Windows CE erstellen. Die Programmierung ist ähnlich der einer Anwendung für einen gewöhnlichen MS Windows PC. Es besteht unter Windows CE lediglich eine Einschränkung der verfügbaren WIN32-API's.

Basisvoraussetzung für die Applikationsprogrammierung

- **Microsoft eMbedded Visual C++ 4.0**
- **SDK** (Software Development Kit) vom Hersteller des Windows CE Gerätes (KEBA).
  Durch den SDK werden der Programmierumgebung die prozessorabhängigen Header- und Bibliotheksdateien der OEM-Anpassung zur Verfügung gestellt.
- Wenn spezielle hardwarenahe Funktionen des KeTop verwendet werden sollen, muss der SDK des KeTop nachinstalliert werden, da in dem standardmäßig mitgelieferte SDK für HPC (Handheld PC) und HPC Pro (Handheld PC Professional) gewisse kundenspezifische Anpassungen noch nicht berücksichtigt sind.

KEBA © KEBA 2006

KeTop API Design

Das gesamte SDK ist in einer einzigen "dynamic link library" (DLL) implementiert. Alle in diesem Dokument beschriebenen Funktionen werden von dieser DLL exportiert.
Das zugehörige SDK befindet sich auf der CD „KeTop SK001“.

Folgende Plattformnamen sind den KeTops zugeordnet:
KeTop T100: KETOPT100
KeTop T50: KETOPT50

- Systemdateien, die vom Visual Studio in das *\windows* Verzeichnis kopiert werden, bleiben nicht dauerhaft gespeichert. Aus diesem Grund müssen diese Dateien manuell in das Verzeichnis *\IPSM\windows* (ist gegebenenfalls vom Benutzer anzulegen) kopiert werden.

  Bei einem Neustart von Windows CE werden diese Dateien automatisch in das *\windows* Verzeichnis kopiert. Diese Dateien stehen somit dem Betriebssystem wieder zur Verfügung und sind auch nach einem Neustart wieder vorhanden.

© KEBA 2006

Gemeinsame Datentypen

Dieser Abschnitt enthält eine genaue Beschreibung der gemeinsamen Datentypen für die Kommunikation mit dem Handterminal. Weitere Informationen, siehe TpuHwDataTypes.h.

INT8	8-Bit ganzzahlige Variable mit Vorzeichen.
UINT8	8-Bit ganzzahlige Variable ohne Vorzeichen.
BacklightStat	Enum, zeigt Zustand der Hintergrundbeleuchtung an.
JoystickPos	Struct, für Joystickdaten.
Status Struct	Beschreibt den Zustand beim Hochlauf des Gerätes.
EventMsg	Enum, beschreibt die erhaltene Ereignismeldung.
EventMsgDomains	Enum, beschreibt für welche Ereignisse ein Handler registriert wurde.
eventCallback	Funktionszeiger auf Callback-Funktion.

Behandlung von Fehlern

Regeln

- Alle Funktionen, die einen Eingangsparameter erwarten, prüfen, ob sich der Parameter innerhalb des Bereiches befindet, und ob er vom richtigen Datentyp ist. Liegt ein Parameter außerhalb des Bereiches, liefert die Funktion INVALID_ARG_RANGE zurück.
- Alle Funktionen, die einen Zeiger für Ausgangsdaten als Parameter erwarten, prüfen, ob es sich um einen gültigen Zeiger handelt, d.h. der Zeiger darf nicht NULL sein. Ist der Zeiger ungültig, liefert die Funktion INVALID_ARG_INVALID_PTR zurück.
- Alle Funktionen, die einen beliebigen String als Parameter haben, prüfen, ob der Zeiger auf den String gültig ist. Ist der Zeiger ungültig, liefert die Funktion INVALID_ARG_INVALID_STR_PTR zurück.

Defines

SUCCESS	0
OK	0
FAIL	1
INVALID_ARG_RANGE	2
INVALID_ARG_PTR	3
INVALID_ARG_STR_PTR	4
INVALID_ARG_UNKNWN_COOKIE	5
INVALID_ARG_UNKNWN_DOMAIN	6
INVALID_NOT_CALIBRATED	7
INVALID_POTI_NOT_CALIBRATED	8
UNSAVE_DLL_TERMINATION	-100
API_NOT_INIT	-101

KEBA © KEBA 2006

Funktionen

Funktionen zum Starten und Beenden

Dieser Abschnitt beschreibt die Funktionen, die für das Starten und Beenden der KetopApi.dll benötigt werden.

KtpAPIInit

Deklaration	`UINT8 KtpAPIInit(void);`
Beschreibung	Diese Methode initalisiert die KetopAPI
Argumente	-

KtpAPIDeinit

Deklaration	`void KtpAPIDeinit(void);`
Beschreibung	Diese Methode macht alle Initalisierungen der KtpAPIInit rückgängig.
Argumente	-

© KEBA 2006

Funktionen zur Konfiguration

Dieser Abschnitt beschreibt die Funktionen, die für die Konfiguration des Gerätes zur Verfügung stehen. Alle Funktionen retournieren als Fehlercode einen der oben angeführten Defines.

KtpSetBrightness

Deklaration	`UINT8 KtpSetBrightness(/*[in]*/UINT8 u8_Brightness);`
Beschreibung	Diese Methode legt die Helligkeit des LC-Displays am Gerät fest.
Argumente	0-7, 0 = min, 7 = max

KtpSetBrightnessPercent

Deklaration	`UINT8 KtpSetBrightnessPercent(/*[in]*/UINT8 u8_Brightness);`
Beschreibung	Diese Methode legt die Helligkeit des LC-Displays am Gerät in Prozent fest.
Argumente	0-100%

KtpSetContrast

Deklaration	`UINT8 KtpSetContrast(/*[in]*/UINT8 u8_Contrast);`
Beschreibung	Diese Methode legt den Kontrast des LC-Displays am Gerät fest.
Argumente	0-31, 0 = min, 31 = max (Ketop T50: 0-63, 0 = min, 63 = max)

KtpSetContrastPercent

Deklaration	`UINT8 KtpSetContrastPercent(/*[in]*/UINT8 u8_Contrast);`
Beschreibung	Diese Methode legt den Kontrast des LC-Displays am Gerät in Prozent fest.
Argumente	0-100%

KtpSwitchBacklight

Deklaration	`UINT8 KtpSwitchBacklight(/*[in]*/BacklightStat backLight);`
Beschreibung	Schaltet die Hintergrundbeleuchtung des LC-Displays am Gerät ein/aus.
Argumente	BACKLIGHT_ON, BACKLIGHT_OFF

KtpSetScreenSaverTimeOutMin

Deklaration	`UINT8 SetScreenSaverTimeOutMin(/*[in]*/UINT8 u8_ScreenSaverTO);`
Beschreibung	Legt den Timeoutwert des Bildschirmschoners in Minuten fest.
Argumente	0-255, 0 = off, 255 = max

KEBA © KEBA 2006

KtpSetScreenSaverTimeOutSec

Deklaration	`UINT8 SetScreenSaverTimeOut-Min(/*[in]*/UINT16u16_ScreenSaverTO);`
Beschreibung	Legt den Timeoutwert des Bildschirmschoners in Sekunden fest.
Argumente	0-65535, 0 = off, 65535 = max

KtpSetBuzzerVolume

Deklaration	`UINT8 KtpSetBuzzerVolume(/*[in]*/UINT8 u8_Volume);`
Beschreibung	Legt die Lautstärke des Summers fest.
Argumente	0-16, 0 = off, 16 = max

© KEBA 2006

Funktionen zum Auslesen der Konfiguration

Diese Funktionen retournieren den aktuellen Wert der Konfigurationsparameter. Keine der Funktionen benötigt Parameter. Diese Funktionen ermöglichen keine Überprüfung auf Fehler, da der Returnwert der Funktion der Wert des Konfigurationsparameters ist.

KtpGetBrightness

Deklaration	`UINT8 KtpGetBrightness(void);`
Beschreibung	Liefert den aktuellen Helligkeitswert des LC-Displays.
Argumente	-

KtpGetBrightnessPercent

Deklaration	`UINT8 KtpGetBrightnessPercent(/*[out]*/UINT8 *u8_Brightness);`
Beschreibung	Liefert den aktuellen Helligkeitswert des LC-Displays in Prozent.
Argumente	UINT8 *u8_Brightness: Helligkeitswert 0-100%

KtpGetContrast

Deklaration	`UINT8 KtpGetContrast(void);`
Beschreibung	Liefert den aktuellen Kontrastwert des LC-Displays.
Argumente	-

KtpGetContrastPercent

Deklaration	`UINT8 KtpSetContrastPercent(/*[out]*/UINT8 *u8_Contrast);`
Beschreibung	Liefert den aktuellen Kontrastwert des LC-Displays in Prozent.
Argumente	UINT8 *u8_Contrast: Kontrastwert 0-100%

KtpGetBacklight

Deklaration	`TKtpBacklightStat KtpGetBacklight(void);`
Beschreibung	Liefert den aktuellen Zustand der Hintergrundbeleuchtung.
Argumente	-

KtpGetScreenSaverTimeoutMin

Deklaration	`UINT8 GetScreenSaverTimeOutMin(void);`
Beschreibung	Liefert den aktuellen Timeoutwert des Bildschirmschoners.
Argumente	-

KtpGetScreenSaverTimeoutSec

Deklaration	`UINT8 GetScreenSaverTimeOutSec(void);`
Beschreibung	Liefert den aktuellen Timeoutwert des Bildschirmschoners.
Argumente	-

KtpGetBuzzerVolume

Deklaration	`UINT8 KtpGetBuzzerVolume(void);`
Beschreibung	Liefert den aktuellen Lautstärkewert des Summers.
Argumente	-

KEBA © KEBA 2006

Peripheriefunktionen

KtpJoystickIsInstalled

Deklaration	`UINT8 KtpJoystickIsInstalled(void);`
Beschreibung	Retourniert die Anzahl der Joystickachsen , falls ein Joystick am Gerät installiert ist, andernfalls 0.
Argumente	-

KtpWheelIsInstalled

Deklaration	`UINT8 KtpWheelIsInstalled(void);`
Beschreibung	Retourniert 1, falls ein Handrad am Gerät installiert ist, andernfalls 0.
Argumente	-

KtpPotiIsInstalled

Deklaration	`UINT8 KtpPotiIsInstalled(void);`
Beschreibung	Retourniert 1, falls ein Override-Potentiometer am Gerät installiert ist, andernfalls 0.
Argumente	-

KtpGetJoystickPos

Deklaration	`UINT8 KtpGetJoystickPos (/*[out]*/TKtpJoystickPos *p_jPos);`
Beschreibung	Liefert die aktuelle Position des Joysticks.
Argumente	TKtpJoystickPos structure. Jede Komponente in der Struktur darf sich nur im Bereich von -15 bis 15 befinden.
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Joystick am Gerät installiert ist. Falls kein Joystick installiert ist, sind die Werte der Komponenten undefiniert.

KtpGetJoystickPosEx

Deklaration	`UINT8 KtpGetJoystickPosEx(/*[out]*/int *posX, int *posY, int *posZ);`
Beschreibung	Liefert die aktuelle Position des Joysticks.
Argumente	Jeder Positionswert darf sich nur im Bereich von -15 bis 15 befinden.
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Joystick am Gerät installiert ist. Falls kein Joystick installiert ist, sind die Werte der Komponenten undefiniert.

KtpGetJoystickPosRaw

Deklaration	`UINT8 KtpGetJoystickPosRaw (/*[out]*/TktpJoystickPosRaw *p_jPos);`
Beschreibung	Liefert die aktuellen Rohdaten des Joysticks.
Argumente	-
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Joystick am Gerät installiert ist. Falls kein Joystick installiert ist, sind die Werte der Komponenten undefiniert.

© KEBA 2006

KtpGetJoystickPosRawEx

Deklaration	`UINT8 KtpGetJoystickPosEx(/*[out]*/UINT16 *posX, UINT16 *posY, UINT16 *posZ);`
Beschreibung	Liefert die aktuellen Rohdaten des Joysticks.
Argumente	-
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Joystick am Gerät installiert ist. Falls kein Joystick installiert ist, sind die Werte der Komponenten undefiniert.

KtpSetJoystickCalibData

Deklaration	`UINT8 KtpSetJoystickCalibData(/*in*/TKtpJoystickChannel ch UINT16 rawMin, UINT16 rawCenter , UINT16 rawMax, UINT16 calibRange);`
Beschreibung	Kalibriert den die Achse des Joysticks Anmerkungen: Diese Funktion darf nur aufgerufen werden, wenn ein Joystick am Gerät installiert ist. Falls kein Joystick installiert ist, sind die Werte der Komponenten undefiniert.
Argumente	TktpJoystickChannel ch: Kanal der kalibriert werden soll. UINT16 rawMin: Wert für den kleinsten Rohwert UINT16 ramCenter: Mittelwert der Rohdaten UINT16 rawMax: Maximalwert des Rohwert UINT16 calibRange: Maximalausschlag des Joysticks

KtpGetOverridePoti

Deklaration	`UINT8 KtpGetOverridePoti(/*out*/UINT8 *p_pos);`
Beschreibung	Liefert den aktuellen Wert des Override-Potentiometers.
Argumente	Zeiger auf Variable, die eine Kopie des aktuellen Wertes des Override-Potentiometers erhält.
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Override-Potentiometer am Gerät installiert ist. Falls kein Override-Potentiometer installiert ist, ist das Ergebnis undefiniert.

KtpGetOverridePotiRaw

Deklaration	`UINT8 KtpGetOverridePotiRaw(/*out*/UINT16 *p_pos);`
Beschreibung	Liefert die unkalibireten Daten des Override-Potentiometers.
Argumente	Zeiger auf Variable, die eine Kopie des aktuellen Wertes des Override-Potentiometers erhält.
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Override-Potentiometer am Gerät installiert ist. Falls kein Override-Potentiometer installiert ist, ist das Ergebnis undefiniert.

KtpGetEnablingDevice

Deklaration	`UINT8 KtpGetEnablingDevice (/*in*/TKtpEnablingDeviceCircuit circuit);`
Beschreibung	Setzt die aktuelle Position des Handrades auf den im val übergebenen Wert
Argumente	val ist der Wert auf den das Handrad gesetzt werden soll
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Handrad am Gerät installiert ist. Falls kein Handrad installiert ist, ist das Ergebnis undefiniert.

KEBA © KEBA 2006

KtpSetPotiCalibData

Deklaration	`UINT8 KtpSetPotiCalibData(/*in*/ UINT16 rawMin, UINT16 rawMax, UINT16 calibRange);`
Beschreibung	Kalibriert das Overridepoti. Diese Funktion darf nur aufgerufen werden, wenn ein Poti am Gerät installiert ist. Falls kein Poti installiert ist, sind die Werte der Komponenten undefiniert.
Argumente	UINT16 rawMin: Wert für den kleinsten Rohwert UINT16 rawMax: Maximalwert des Rohwert UINT16 calibRange: Maximalausschlag des Overridepotis

KtpGetWheelValue

Deklaration	`UINT8 KtpGetWheelValue(/*out*/UINT16 *p_val);`
Beschreibung	Liefert den aktuellen Wert des Handrades.
Argumente	Zeiger auf Variable, die eine Kopie des aktuellen Wertes des Handrades erhält.
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Handrad am Gerät installiert ist. Falls kein Handrad installiert ist, ist das Ergebnis undefiniert.

KtpSetWheelValue

Deklaration	`UINT8 KtpSetWheelValue(/*in*/UINT16 val);`
Beschreibung	Setzt die aktuelle Position des Handrades auf den im val übergebenen Wert
Argumente	val ist der Wert auf den das Handrad gesetzt werden soll
Anmerkung	Diese Funktion darf nur aufgerufen werden, wenn ein Handrad am Gerät installiert ist. Falls kein Handrad installiert ist, ist das Ergebnis undefiniert.

LED-Funktionen

Die für nachfolgende Funktionen benötigte LED-Nummerierung (ledNr) kann dem Kap. "LED-Nummerierung" entnommen werden.

KtpSetKeyboardLed

Deklaration	`UINT8 KtpSetKeyboardLed(/*in*/UINT8 ledNr, TLedState ledState);`
Beschreibung	Schaltet die angegebene LED in den gewünschten Zustand.
Argumente	UINT8 ledNr: Nummer der zu setzenden LED. UINT ledState: Zustand, auf den die LED gesetzt wird.

KtpGetKeyboardLed

Deklaration	`TLedState KtpGetKeyboardLed(/*in*/UINT8 ledNr);`
Beschreibung	Retourniert den Zustand der gewünschten LED.
Argumente	UINT8 ledNr: Nummer der LED.

© KEBA 2006

Diverse Funktionen

Diese Funktionen führen diverse Aktionen am Gerät aus.

KtpGetStatus

Deklaration	`UINT8 KtpGetStatus(/*[out]*/Status *p_tpuStatus);`
Beschreibung	Liefert den Zustand beim Hochlauf des Handterminals.
Argumente	Status `*p_tpuStatus`, Zeiger auf Speicherbereich, der eine Kopie des Hochlaufzustandes erhält.

KtpWriteToFlash

Deklaration	`UINT8 WriteToFlash(void);`
Beschreibung	Schreibt die Registry des Gerätes ins Flash.
Argumente	-

KtpReset

Deklaration	`UINT8 KtpReset(void);`
Beschreibung	Startet das Gerät neu
Argumente	-

KtpGetVersionString

Deklaration	`UINT8 KtpGetVersionString(/*out*/TCHAR *wszVersionString, unsigned int bufferLen);`
Beschreibung	Speichert die Versionsbezeichnung im übergebenen String ab
Argumente	TCHAR *wszVersionString: Zeiger auf den Buffer für den Versions-string unsigned int bufferLen: Länge des Übergebenen Buffers

KtpGetEEPromData

Deklaration	`UINT8 KtpGetEEPromData(/*out*/TEEPromData *data);`
Beschreibung	Liest die Daten aus dem EEProm aus
Argumente	TEEPromData data: Datenstrucktur für die Daten im EEProm

KtpWriteByteToEEProm

Deklaration	`UINT8 KtpWriteByteToEEProm (/*in*/UINT16 addr, UINT8 data);`
Beschreibung	Schreibt ein Byte auf die in addr übergeben Speicherstelle des EEProms
Argumente	UINT16 addr: Adresse der Speicherstelle UINT8 data: Daten zum Speichern

KtpReadByteFromEEProm

Deklaration	`UINT8 KtpReadByteFromEEProm (/*in*/UINT16 addr, /*out*/ UINT8 *pData);`
Beschreibung	Liest ein Byte von der in addr übergeben Speicherstelle des EEProms
Argumente	UINT16 addr: Adresse der Speicherstelle UINT8 *pData: Daten aus dem EEProm

KEBA © KEBA 2006

KtpLaunchTouchScreenCalibApp

Deklaration	`UINT8 KtpLaunchTouchScreenCalibApp (void);`
Beschreibung	Startet das Touchscreenkalibriertool
Argumente	-

KtpGetTemperature

Deklaration	`UINT8 KtpGetTemperature (void);`
Beschreibung	Liefert die aktuelle Temperatur des KeTops.
Argumente	-

KtpPlaySound

Deklaration	`UINT8 KtpPlaySound (/*in*/ UINT16 soundNr);`
Beschreibung	Spielt den in soundNr übergebenen Systemsound ab.
Argumente	UINT16 soundNr: Nummer des Systemsounds.

KtpDoBeep

Deklaration	`UINT8 KtpDoBeep (/*in*/ UINT8 beepTime10ms);`
Beschreibung	Piepst für beepTime10ms * 10 ms
Argumente	UINT8 beeptime10ms: Dauer des Beeps in Anzahl * 10ms

KtpGetVariantData

Deklaration	`UINT8 KtpGetVariantData (/*out*/ TKtpVariantData *data);`
Beschreibung	Liest die Gerätekonfigruation aus dem EEProm
Argumente	TKtpVariantData data: Datenstrucktur für die VariantData

Funktionen zur Registrierung von Ereignissen

Diese Funktionen registrieren/deregistrieren Callback-Funktionen für verschiedene Ereignisse. Als Ereignisse können Joystick, Override-Poti, Handrad und Tastatur verwendet werden.

KtpInstallWheelEventCallback

Deklaration	`UINT8 KtpInstallWheelEventCallback (/*in*/TktpWheelEventCallback pWheelProc, int *cookie);`
Beschreibung	Registert eine Callback-Funktion für das WheelEvent und liefert einen Index (cookie) für die Callback-Funktion
Argumente	TKtpWheelEventCallback pWheelProc: Callback-Function die aufgerufen werden soll wenn der Event auftritt. int cookie: Index für die Callback-Funktion wird zum Entfernen der Callback-Funktion benötigt.

KtpInstallWheelEventMessage

Deklaration	`UINT8 KtpInstallWheelEventMessage (HWND hWnd, int *cookie);`
Beschreibung	Registert einen WindowHandler für das WheelEvent und liefert einen Index
Argumente	HWND hWnd: WindowHandler, an den die Nachricht gesendet wird. int cookie: Index, wird zum Entfernen des WindowHandlers benötigt.

© KEBA 2006

KtpRemoveWheelEventCallback

Deklaration	`UINT8 KtpRemoveWheelEventCallback (int cookie);`
Beschreibung	Entfernt die WheelEventCallback-Funktion.
Argumente	int cookie: Index für die zu entfernende Callback-Funktion

KtpRemoveWheelEventMessage

Deklaration	`UINT8 KtpRemoveWheelEventMessage (int cookie);`
Beschreibung	Entfernt den WindowHandler.
Argumente	int cookie: Index für den zu entfernenden WindowHandler

KtpInstallOvrEventCallback

Deklaration	`UINT8 KtpInstallOvrEventCallback (/*in*/ TKtpOvrEventCallback pOvrProc, /*out*/ int *cookie);`
Beschreibung	Registert eine Callback-Funktion für das OverrideEvent und liefert einen Index (cookie) für die Callback-Funktion
Argumente	TKtpOvrEventCallback pOvrProc: Callback-Function die aufgerufen werden soll wenn der Event auftritt. int cookie: Index für die Callback-Funktion wird zum Entfernen der Callback-Funktion benötigt.

KtpInstallOvrEventMessage

Deklaration	`UINT8 KtpInstallOvrEventMessage (HWND hWnd, int *cookie);`
Beschreibung	Registert einen WindowHandler für das OverrideEvent und liefert einen Index
Argumente	HWND hWnd: WindowHandler, an den die Nachricht gesendet wird. int cookie: Index, wird zum Entfernen des WindowHandlers benötigt.

KtpRemoveOvrEventCallback

Deklaration	`UINT8 KtpRemoveOvrEventCallback (int cookie);`
Beschreibung	Entfernt die OverrideEventCallback-Funktion.
Argumente	int cookie: Index für die zu entfernende Callback-Funktion

KtpRemoveOvrEventMessage

Deklaration	`UINT8 KtpRemoveOvrEventMessage (int cookie);`
Beschreibung	Entfernt den WindowHandler.
Argumente	int cookie: Index für den zu entfernenden WindowHandler

KtpInstallKbdEventCallback

Deklaration	`UINT8 KtpInstallKbdEventCallback (/*in*/ TktpKbdEventCallback pKbdProc, /*out*/ int *cookie);`
Beschreibung	Registert eine Callback-Funktion für das KeyboardEvent und liefert einen Index (cookie) für die Callback-Funktion
Argumente	TktpKbdEventCallback pKbdProc: Callback-Function die aufgerufen werden soll wenn der Event auftritt. int cookie: Index für die Callback-Funktion wird zum Entfernen der Callback-Funktion benötigt.

KEBA © KEBA 2006

KtpInstallKbdEventMessage

Deklaration	`UINT8 KtpInstallKbdEventMessage (HWND hWnd, int *cookie);`
Beschreibung	Registert einen WindowHandler für das KeyboardEvent und liefert einen Index
Argumente	HWND hWnd: WindowHandler, an den die Nachricht gesendet wird. int cookie: Index, wird zum Entfernen des WindowHandlers benötigt.

KtpRemoveKbdEventCallback

Deklaration	`UINT8 KtpRemoveKbdEventCallback (int cookie);`
Beschreibung	Entfernt die WheelEventCallback-Funktion.
Argumente	int cookie: Index für die zu entfernende Callback-Funktion

KtpRemoveKdbEventMessage

Deklaration	`UINT8 KtpRemoveKbdEventMessage (int cookie);`
Beschreibung	Entfernt den WindowHandler.
Argumente	int cookie: Index für den zu entfernenden WindowHandler

KtpInstallJoyEventCallback

Deklaration	`UINT8 KtpInstallJoyEventCallback (/*in*/ TktpJoyEventCallback pJoyProc, /*out*/ int *cookie);`
Beschreibung	Registert eine Callback-Funktion für das JoystickEvent und liefert einen Index (cookie) für die Callback-Funktion
Argumente	TktpJoyEventCallback pJoyProc: Callback-Function die aufgerufen werden soll wenn der Event auftritt. int cookie: Index für die Callback-Funktion wird zum Entfernen der Callback-Funktion benötigt.

KtpInstallJoyEventMessage

Deklaration	`UINT8 KtpInstallJoyEventMessage (HWND hWnd, int *cookie);`
Beschreibung	Registert einen WindowHandler für das JoystickEvent und liefert einen Index
Argumente	HWND hWnd: WindowHandler, an den die Nachricht gesendet wird. int cookie: Index, wird zum Entfernen des WindowHandlers benötigt.

KtpRemoveJoyEventCallback

Deklaration	`UINT8 KtpRemoveJoyEventCallback (int cookie);`
Beschreibung	Entfernt die JoyEventCallback-Funktion.
Argumente	int cookie: Index für die zu entfernende Callback-Funktion

KtpRemoveJoyEventMessage

Deklaration	`UINT8 KtpRemoveJoyEventMessage (int cookie);`
Beschreibung	Entfernt den WindowHandler.
Argumente	int cookie: Index für den zu entfernenden WindowHandler

© KEBA 2006

KtpLaunchJoystickCalibApp

Deklaration	`UINT8 KtpLaunchJoystickCalibApp(HANDLE *pProcHandle);`
Beschreibung	Starte das Programm zum Kalibrieren des Joysticks
Argumente	HANDLE *pProcHandle: Handle für das gestartete Programm

KtpGetDisplayRotation

Deklaration	`TKtpDispalyRot KtpGetDisplayRotation (void);`
Beschreibung	Liefert die Einstellung für die DisplayRotation (für links oder rechtshänder).
Argumente	-

KtpSetDisplayRotation

Deklaration	`UINT8 KtpSetDisplayRotation(TKtpDisplayRot dispRot);`
Beschreibung	Ändert die Darstellung am Display für links oder Rechtshänder. Kann bei einigen Geräten keine Auswirkung haben.
Argumente	TKtpDisplayRot dispRot: eKtpDisplayLeft für Linkshänder, eKtpDisplayRight für Rechtshänder

KtpEraseRegistry

Deklaration	`UINT8 KtpEraseRegistry(void);`
Beschreibung	Löscht beim nächsten Restart die Registry des Gerätes
Argumente	-

KtpErasePSM

Deklaration	`UINT8 KtpErasePSM(void);`
Beschreibung	Löscht beim nächsten Restart das Flash File System des Gerätes
Argumente	-

KtpGetPowerFailState

Deklaration	`UINT8 KtpGetPowerFailState(void);`
Beschreibung	Liefert den aktuellen Zustand des PowerFail Einganges.
Argumente	-

KtpForcePressedHardbuttons

Deklaration	`UINT8 KtpForcePressedHardbuttons (void);`
Beschreibung	Wird diese Funktion im System aufgerufen, wird eine WM_KEYDOWN Message für jedes gedrückte Schaltelement versendet und die KeyboardCallback-Funktion aufgerufen. Diese Funktion wird auch beim Initalisieren der KetopApi und beim Installieren der KeyboardCallback-Funktion aufgerufen.
Argumente	-

KEBA © KEBA 2006

KtpIsJoystickCalibrated

Deklaration	`UINT8 KtpIsJoysstickCalibrated (void);`
Beschreibung	Liefert OK (0) wenn alles Joystickachsen kalibriert sind. Falls eine oder mehre Achsen nicht kalibriert ist / sind wird INVALID_NOT_CALIBRATED (7) zurückgegeben. Ist kein Joystick vorhanden kommt als Ergebnis INVALID_NOT_SUPPORTED (6) zurück.
Argumente	-

KtpIsPotiCalibrated

Deklaration	`UINT8 KtpIsPotiCalibrated (void);`
Beschreibung	Liefert OK (0) wenn das Override-Potentiometer kalibiriert ist. Falls das Override-Potentiometer nicht kalibriert ist wird INVALID_NOT_CALIBRATED (7) zurückgegeben. Ist kein Override-Potentiometer vorhanden kommt als Ergebnis INVALID_NOT_SUPPORTED (6) zurück.
Argumente	-

KtpRestoreMemorySettings

Deklaration	`UINT8 KtpRestoreMemorySettins(void);`
Beschreibung	Speichert den in der Registry unter "System\MemorySettings\ StorePages" angegeben Wert als Storage Memory. Wenn der Wert aus der Registry sich in Bereichen befinden wo Speicher für Storage bzw Program Memory reserviert ist werden die Settings nicht verändert und INVALID_ARG_RANGE zurückgegeben. Ist das Speichern der Memorysettings nicht erlaubt ("System\MemorySettings\MemorySaveEnable"=0) wird INVALID_NOT_SUPPORTED als Rückgabewert geliefert.
Argumente	-

KtpStoreCurrentMemorySettings

Deklaration	`UINT8 KtpStoreCurrentMemorySettins(void);`
Beschreibung	Speichert die Memoryeinstellung in der Registry ab ("System\MemorySettings\StorePages") falls das Speichern erlaubt ist. Ist das Speichern nicht erlaubt wird INVALID_NOT_SUPPORTED zurückgegeben.
Argumente	-

KtpShowInputPanel

Deklaration	`UINT8 KtpShwoInputPanel(UINT8 show);`
Beschreibung	Öffnet (show=1) oder schließt (show=0) das Inputpanel am Display und gibt als Rückgabewert OK oder FAIL aus.
Argumente	-

© KEBA 2006

Update API Design

Alle für ein Image Update benötigten Methoden sind in einer einzigen "dynamic link library" implementiert (update.dll). Die in diesem Kapitel beschriebenen Funktionen werden von dieser Datei zur Verfügung gestellt. Um die update.dll zu betreiben benötigt man die passende KetopAPI.dll. Beide *.dll – Dateien müssen im Image vorhanden sein und müssen von der Version zusammenpassen.

Behandlung von Fehlern

Regeln

- Alle Funktionen, die einen Eingangsparameter erwarten, prüfen, ob sich der Parameter innerhalb des Bereiches befindet und ob er vom richtigen Datentyp ist. Liegt ein Parameter außerhalb des Bereiches, liefert die Funktion ERROR_INVALIDE_RANGE zurück.

Defines

OK	0
SUCCESS	0
FAIL	1
ERROR_INVALIDE_RANGE	100
ERROR_IMGUPD_INIT	101
ERROR_IMGUPD_FILEOPEN	102
ERROR_IMGUPD_WRONGIMG	103
ERROR_IMGUPD_PROGRAMERROR	104
ERROR_IMGUPD_TOMUCHUPD	105
ERROR_PROGRAM_FLASH_BURN	106
ERROR_PROGRAM_FLASH_ERASE	107
ERROR_IMGUPD_NOTEQUAL	108
ERROR_IMGUPD_FILEWRITE	109
ERROR_IMGUPD_FILEREAD	110
ERROR_WRONG_FILE_HEADER	111

KEBA © KEBA 2006

Initialisierung

Alle für den Betrieb notwendigen Initialisierungen werden mit dem Starten bzw. dem Laden der update.dll durchgeführt.

Funktionen

UpdStartImageUpdate

Deklaration	`BOOL UpdStartImageUpdate (LPCTSTR fileName, BOOL eraseRegPSM, TUpdProgressCallback pCallback)`
Beschreibung	Mit dieser Funktion wird das in `fileName` übergeben File in den FLASH-Speicher des Gerätes geschrieben. Ist `eraseRegPSM` gesetzt wird nach erfolgreichen schreiben des FLASH wird ein löschen der Registry bzw des PSM eingeleitet. Wird mit `pCallback` eine Funktion übergeben, wird dies in Abhängigkeit des Programmierfortschrittes aufgerufen. Als Rückgabe wert liefert die Funktion OK oder den entsprechenden Fehlerwert
Argumente	`LPCTSTR fileName:` Name des Image Files `BOOL eraseRegPSM:` Flag zum Löschen der Registry bzw des PSM `TUpdProgressCallback pCallback:` Callback Funktion für den Programmierfortschritt oder 0

UpdPartialImageUpdate

Deklaration	`BOOL UpdPartialImageUpdate (LPCTSTR fileName, unsigned long from, unsigned long to, BOOL eraseRegPSM, TUpdProgressCallback pCallback)`
Beschreibung	Mit dieser Funktion wird das in `fileName` übergeben File von der Adresse `from` bis zur Adresse to in den FLASH-Speicher des Gerätes geschrieben. Ist eraseRegPSM gesetzt wird nach erfolgreichen schreiben des FLASH wird ein löschen der Registry bzw des PSM eingeleitet. Wird mit `pCallback` eine Funktion übergeben, wird diese in Abhängigkeit des Programmierfortschrittes aufgerufen. Als Rückgabe wert liefert die Funktion OK oder den entsprechenden Fehlerwert
Argumente	`LPCTSTR fileName:` Name des Image Files `unsigned long from:` Erste Adresse die Überschrieben werden soll `unsigned long to:` Adresse die nicht mehr überschrieben wird BOOL eraseRegPSM: Flag zum Löschen der Registry bzw des PSM TUpdProgressCallback pCallback: Callback Funktion für den Programmierfortschritt oder 0

© KEBA 2006

UpdCheckFile

Deklaration	`BOOL UpdCheckFile(LPCTSTR fileName, TupdProgressCallback pCallback)`
Beschreibung	`UpdCheckFile` überprüft die Integrität des Image-Files und ob das Image-File zum für das Gerät bestimmt ist. Wird mit `pCallback` eine Funktion übergeben, wird diese in Abhängigkeit des Überprüfungsfortschrittes aufgerufen. Als Rückgabe wert liefert die Funktion OK oder den entsprechenden Fehlerwert
Argumente	`LPCTSTR fileName:` Name des Image Files `TUpdProgressCallback pCallback:` Callback Funktion für den Programmierfortschritt oder 0

UpdCompareFile

Deklaration	`BOOL UpdCompareFile(LPCTSTR fileName, TUpdProgressCallback pCallback);`
Beschreibung	`UpdCompareFile` vergleicht das übergeben Image-File mit dem Inhalt des FLASH-Speichers. Wird mit `pCallback` eine Funktion übergeben, wird diese in Abhängigkeit des Überprüfungsfortschrittes aufgerufen. Als Rückgabe wert liefert die Funktion OK oder den entsprechenden Fehlerwert
Argumente	`LPCTSTR fileName:` Name des Image Files `TUpdProgressCallback pCallback:` Callback Funktion für den Programmierfortschritt oder 0

UpdSetFlashLock

Deklaration	`BOOL UpdSetFlashLock(long from, long to);`
Beschreibung	Mit der Funktion `UpdSetFlashLock` werden die Lock-Bits der FLASH-Bausteine von der Adresse `from` bis zu Adresse `to` gesetzt, und so verhindert das dieser Adreßbereich beschrieben wird. Nach einem Imageupdate sind die Lock-Bits zurückgesetzt. Es wird immer für einen ganzen FLASH-Block das Lock-Bit gesetzt. Der geschützte Bereich kann daher von den übergebenen Adressen abweichen.
Argumente	`long from:` Adresse ab der die Lock-Bits gesetzt werden. `long to:` Adresse bis zu der die Lock-Bits gesetzt werden.

UpdResetFlashLock

Deklaration	`BOOL UpdResetFlashLock(long from, long to);`
Beschreibung	Mit der Funktion `UpdResetFlashLock` werden die Lock-Bits der FLASH-Bausteine von der Adresse `from` bis zu Adresse `to` zurückgesetzt, Es wird immer für einen ganzen FLASH-Block das Lock-Bit zurückgesetzt. Der ungeschützte Bereich kann daher von den übergebenen Adressen abweichen.
Argumente	`long from:` Adresse ab der die Lock-Bits zurückgesetzt werden. `long to:` Adresse bis zu der die Lock-Bits zurückgesetzt werden.

KEBA © KEBA 2006

UpdGetImage

Deklaration	`BOOL UpdGetImage(LPCTSTR fileName, unsigned lont from, unsigned long to, TUpdProgressCallback pCallback);`
Beschreibung	`UpdGetImage` erzeugt ein Abbild des Geräte-Image von der Adresse `from` bis zur Adresse `to` und speichert es im File fileName ab. Wird mit `pCallback` eine Funktion übergeben, wird diese in Abhängigkeit des Fortschrittes aufgerufen. Als Rückgabe wert liefert die Funktion OK oder den entsprechenden Fehlerwert
Argumente	`LPCTSTR fileName:` Name des Image Files `unsigned long from:` Startadresse die ins File geschrieben wird `unsigned long to:` Adresse die nicht mehr gesichert wird `TUpdProgressCallback pCallback:` Callback Funktion für den Programmierfortschritt oder 0

UpdEraseRegistry

Deklaration	`BOOL UpdEraseRegistry();`
Beschreibung	Mit dem Aufruf von `UpdEraseRegistry` wird beim nächsten Hochlauf die Registry des Gerätes gelöscht.

UpdErasePSM

Deklaration	`BOOL UpdErasePSM();`
Beschreibung	Mit dem Aufruf von `UpdErasePSM` wird beim nächsten Hochlauf der PSM des Gerätes gelöscht.

UpdResetDevice

Deklaration	`BOOL UpdResetDevice();`
Beschreibung	Mit der Funktion `UpdResetDevice` wird das Geräte resetiert.

UpdGetFileVersion

Deklaration	`BOOL UpdGetFileVersion(LPCTSTR fileName, LPTSTR version);`
Beschreibung	`UpdGetFileVersion` gibt die im Image-File gespeicherte Versionsinformation im String `version` zurück. Die Anzahl der Zeichen die in zurückgegeben werden können ist in `VERSION_STRING_LEN` definiert. `version` muß daher größer als `VERSION_STRING_LEN` sein.
Argumente	`LPCTSTR fileName:` Name des Image Files `LPTSTR version:` String für die Versionsinformation

UpdGetImageVersion

Deklaration	`BOOL UpdGetImageVersion(LPTSTR version);`
Beschreibung	`UpdGetImageVersion` gibt die in der Registry gespeicherte Versionsinformation im String `version` zurück. Die Anzahl der Zeichen die in `version` zurückgegeben werden können ist in `VERSION_STRING_LEN` definiert. `version` muß daher größer als `VERSION_STRING_LEN` sein.
Argumente	`LPTSTR version:` String für die Versionsinformation

© KEBA 2006

TupdProgressCallback

Deklaration	`typedef void (* TUpdProgressCallback) (int percent);`
Beschreibung	`TUpdProgressCallback` ist der Prototyp für die Callback-Funktionen, die aus der Update API aufgerufen werden können.
Argumente	`int percent:` Zahl zwischen 0 und 100 die den aktuellen Fortschritt darstellt

KEBA © KEBA 2006

Programm zum Starten von Applikation und KeTop API

In diesem Kapitel wird das Programm zum Starten von Applikation und der KeTop API beschrieben. Das Programm befindet sich als StartAPI.exe im Windowsverzeichnis des KeTops.

Funktionsbeschreibung

Das StartAPI – Programm wird durch einen Eintrag im Startup-Verzeichnis bzw. durch einen Eintrag in der Registry unter [HKEY_LOCAL_MACHINE\init] beim Hochfahren des KeTops gestartet.

Beim Starten des Programms wird zuerst die KeTop API initialisiert und anschließend die Programme die unter [HKEY_LOCAL_MACHINE\Autostart] eingetragen sind gestartet.

Das Programm verbleibt unsichtbar im Speicher und handelt diverse Eingabegeräte
z.B.: Overridepoti und Handrad

Registryeinträge

Unter dem [HKEY_LOCAL_MACHINE\Autostart] sind die Einträge die vom Programm ausgelesen werden.

Es sind folgende Einträge möglich

- **Startx**
  ist ein String, der das zu startende Programm bezeichnet, wobei x eine Zahl zwischen 1 und 255 ist.
- **Paramsx**
  ist ein String, der die Parameter für das in **Startx** bezeichnete Programm enthält. x muss den gleichen Wert wie bei **Startx** haben.
- **Delayx**
  ist ein DWORD und enthält die Wartezeit in Millisekunden bis das nächste Programm gestartet wird. x muss den gleichen Wert wie bei **Startx** haben.
- **DependStartx**
  ist ein Array aus 10 Bytes in dem die Nummern der Programme stehen, die bereits gestartet sein müssen damit das Programm x starten darf. Fehlt dieser Eintrag oder sind alle Bytes 0 so startet das Programm x ohne das andere Programme gestartet sein müssen.
- **DependEndx**
  ist ein Array aus 10 Byte in dem die Nummern der Programme stehen, die beendet werden müssen bevor das Programm x startet. Fehlt dieser Eintrag oder sind alle Bytes 0 so startet das Programm x ohne dass auf andere Programme gewartet wird.

© KEBA 2006

- **StartTypex**
  ist ein DWORD und beeinflusst das Startverhalten der Anwendung.
  StartTypex = 0: Die Anwendung wird automatisch gestartet
  StartTypex = 1: Die Anwendung wird nicht gestartet
  StartTypex = 2: Es erscheint ein Auswahlfenster mit der Meldung „start programm <Yes> / <No>“. Der Bediener kann nun entscheiden, ob die Anwendung gestartet werden soll oder nicht.
  x muss den gleichen Wert wie bei **Startx** haben.

Beispiel

Registry-Auszug des KeTop T100 Image Vers. 2.23a:
(Anzeige-Format wie im Pocket Registry-Editor)

[HKEY_LOCAL_MACHINE\Autostart]

```
Start1          = "DirCopy.exe"
Params1         = "\ipsm\Windows \windows"
Delay1          = 0x00000100 (256)
StartType1      = 0x00000000 (0)

Start10         = "explorer.exe"
Params10        = ""
DependEnd10     = 01 00 00 00 00 00 00 00 00 00
DependStart10   = 00 00 00 00 00 00 00 00 00 00
StartType10     = 0x00000000 (0)

Start20         = "SetTime.exe"
Params20        = "/onstartup"
Delay20         = 0x00000000 (0)
StartType20     = 0x00000000 (0)

Start30         = "cmd.exe"
Params30        = "/c \ipsm\copyfiles.cmd"
Delay30         = 0x00000100 (256)
StartType30     = 0x00000000 (0)
```

Als erstes wird der Eintrag bei "Start1" bearbeitet.
Der Aufruf von "DirCopy.exe \ipsm\Windows \windows" bewirkt, dass alle Unterverzerzeichnisse und Dateien aus dem Verzeichnis \ipsm\windows in das Verzeichnis \windows kopiert werden.

Da jede Windows-Anwendung in einem eigenen Fenster abläuft, können alle "Start.."-Einträge zur selben Zeit ablaufen.

Der Eintrag "Delay1" verzögert die Anwendung "explorer.exe" hier um mindestens 256ms.
Außerdem bewirkt der Eintrag "DependEnd10 = 01 …" eine Abhängigkeit von der vorherigen Anwendung. Diese bewirkt, dass "explorer.exe" erst dann gestartet, wenn die Anwendung "DirCopy.exe" beendet wurde.

KEBA © KEBA 2006

"Start20 = SetTime.exe" bewirkt den Aufruf des Eingabefensters zum setzen von Datum und Uhrzeit.

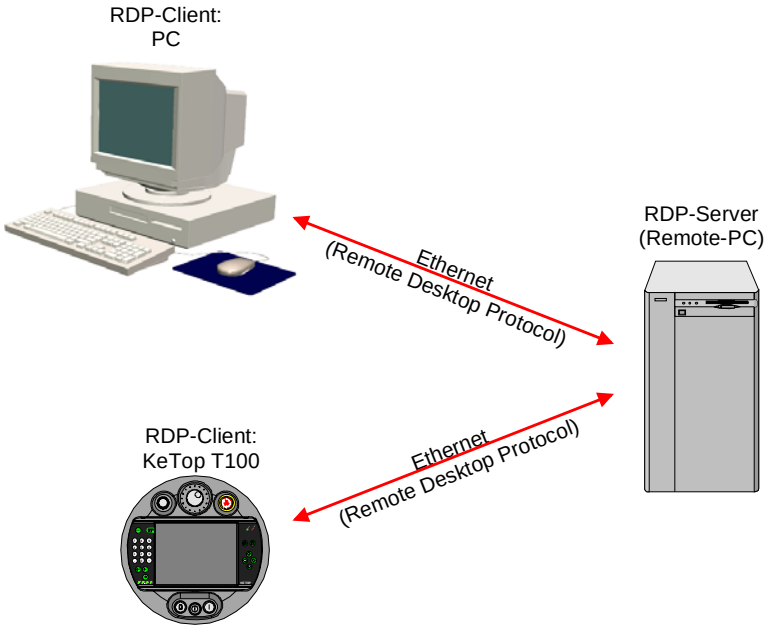
"Start30" ist ein Beispiel für den Aufruf einer eigenen Batchdatei.
Ist die Datei "\ipsm\copyfiles.cmd" nicht vorhanden, dann kann dieser Punkt gelöscht werden.
Sind keine Einträge mehr vorhanden wird das Starten der Applikationen beendet.
Von der StartAPI gestartete Programme werden nicht beendet.

© KEBA 2006

RDP – Verbindung via Remote Desktop Protokoll

Das Remote Desktop Protokoll (Abkürzung „RDP“) ist eine Spezifikation der Firma Microsoft zur Fernbedienung von Anwendungen.
Die frühere Bezeichnung „Microsoft Terminal Server Client“ (Abkürzung „MSTSC“) wurde seitens Microsoft durch die Bezeichnung RDP-Client ersetzt.
Eine umfangreiche Beschreibung steht bei Microsoft im Internet, sowie in der Windows-Hilfe mit den Suchbegriffen „mstsc“ und „rdp“.

Funktionsweise (stark vereinfacht):



RDP-Client <-> RDP-Server

Programm und Daten liegen auf einem entfernten Rechner. Dort erfolgt die Programmausführung und der Datenzugriff. Dieser entfernte Rechner (Remote PC) wird als „RDP-Server“ bezeichnet.

Die Eingabe von Daten und die Anzeige am Bildschirm erfolgt hingegen lokal am sogenannten „RDP-Client“.

Der Datenaustausch zwischen Client und Server erfolgt über das „Remote Desktop Protokoll“.

RDP-Server (Remote-PC)

Der RDP-Server (Remote-PC) benötigt eines der folgenden Betriebssysteme:

KEBA © KEBA 2006

- Windows NT4 Terminalserver
- Windows 2000 Server
- Windows 2003 (Server)
- Windows XP Professional (nicht Windows XP Home)
- Windows XP embededded (Die Server-Funktionalität ist integrierbar.)

Hinweis
Bei Windows XP Professional muss der RDP-Zugriff am PC freigeschaltet sein unter: „Systemsteuerung -> System -> Remote“
(Die Option „(x) Remotedesktop“ muss aktiviert sein.)

RDP-Client (PC)

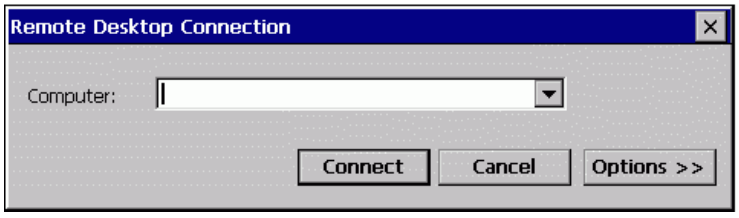
Zum Start des RDP-Client am PC muss die Datei <mstsc.exe> aufgerufen werden. Bei Windows XP steht diese im Windows-Verzeichnis oder unter „Start -> Alle Programme -> Zubehör -> Kommunikation -> Remote-desktopverbindung“.
Für andere Betriebsysteme wie Windows 95/98/NT finden sich die passenden Dateien bei Microsoft im Internet.

RDP-Client (KeTop)

Zum Start des Client am KeTop muss die Datei `cetsc.exe` aufgerufen werden. Sie steht im Windows-Verzeichnis und kann aufgerufen werden über „Start -> Programs -> Remote Desktop Connection“.

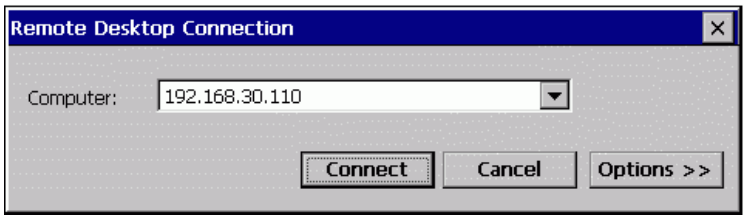
Manueller Start einer RDP-Verbindung

▶ Starten Sie den RDP-Client mittels der Datei `cetsc.exe` und folgendes Anmeldefenster erscheint:



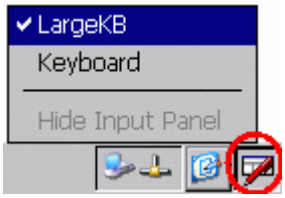
▶ Geben Sie im Feld „Computer“ die IP-Adresse des Ziel-Rechners ein:

© KEBA 2006

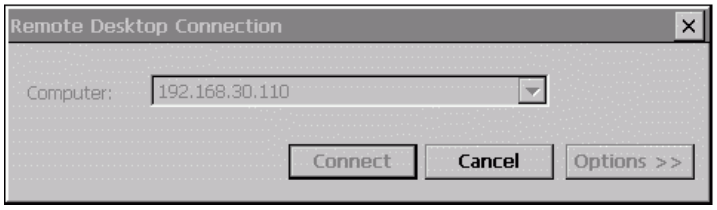


Soll anstelle der IP-Adresse der Netzwerkname verwendet werden, dann muss eine Namensauflösung via DNS-Server oder per Eintrag in der Datei „hosts.ini“ gewährleistet sein.

Hinweis
An dieser Stelle sollte am KeTop das „Input Panel“ eingeblendet werden, denn während des Anmeldevorgangs ist dieser Zugriff gesperrt:



► Klicken Sie auf den Button „Connect“ und der Verbindungsaufbau zum Zielrechner wird eingeleitet:



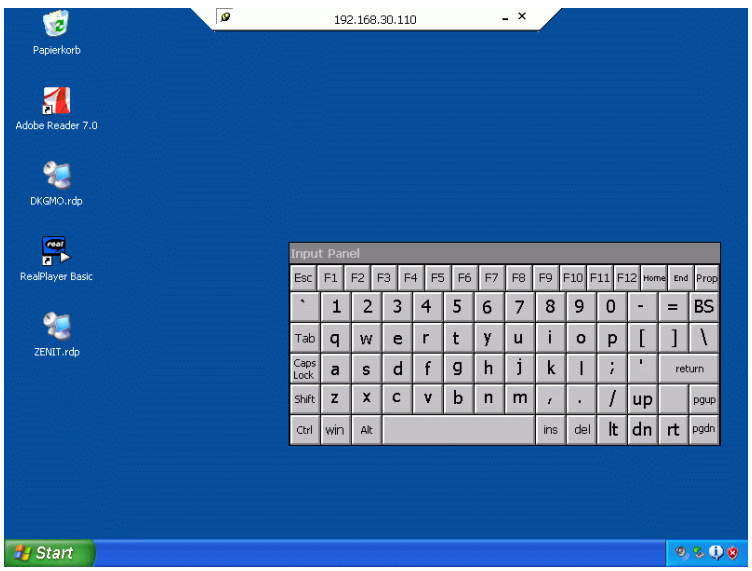
Durch Drücken des Buttons „Cancel“ kann der Verbindungsaufbau unterbrochen werden.

Bei erfolgreicher Herstellung der Verbindung erscheint das Anmeldefenster des Zielrechners mit einer trapezförmigen Titelzeile als Kennzeichen der RDP-Verbindung.

KEBA © KEBA 2006

- Geben Sie zur Anmeldung den Benutzernamen und das Kennwort ein und klicken Sie anschließend auf OK:

Nach erfolgreicher Anmeldung erscheint der Windows Bildschirm des Zielrechners.

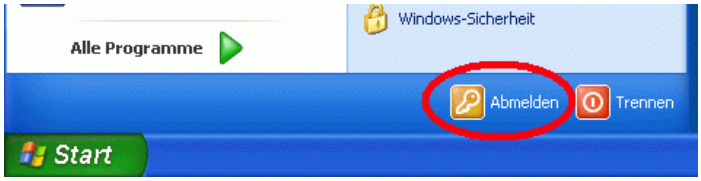


Sollte die trapezförmige Titelzeile in einer Anwendung stören, dann kann sie für den nächsten Anmeldevorgang deaktiviert werden durch klick auf das Symbol links oben.

© KEBA 2006

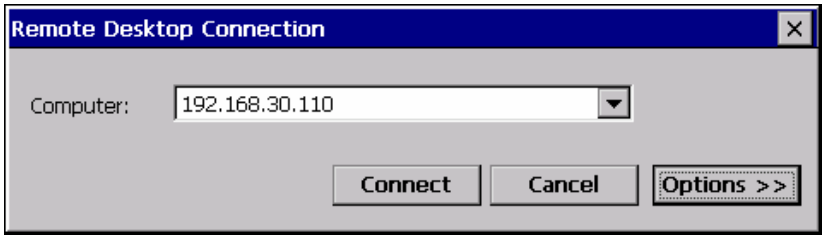
Bei Anzeige dieses Symbols verschwindet die Titelzeile automatisch nach dem nächsten Anmeldevorgang.

Die Abmeldung vom Zielsystem erfolgt durch Aufruf von „Start->Abmelden“.

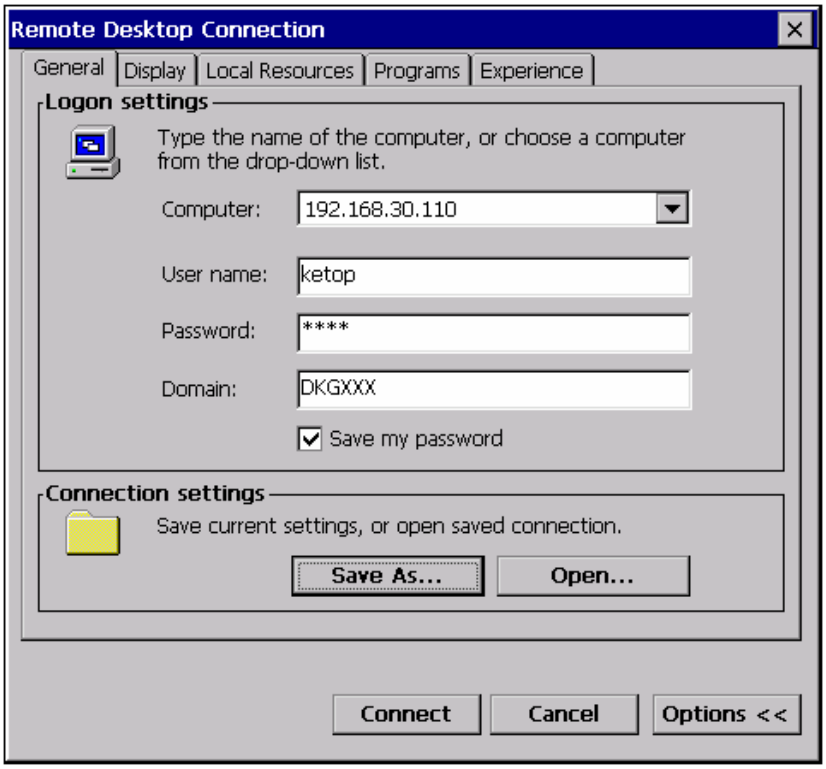


KEBA © KEBA 2006

Speichern von Einstellungen einer RDP-Verbindung

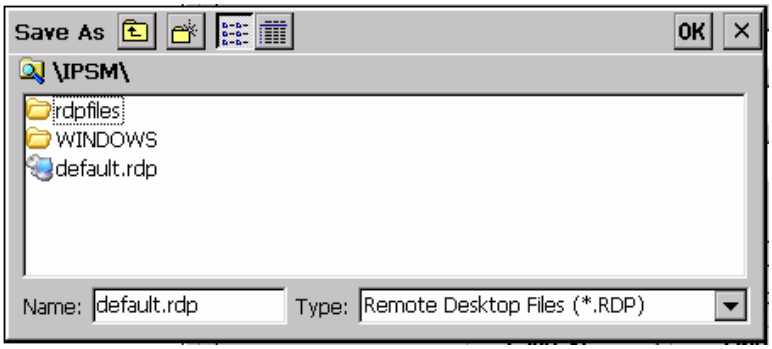


- Klicken Sie im Anmeldefenster des RDP-Client auf den Button „Options >>“ und das Fenster wird auf eine Darstellung mit den aktuellen Einstellungen erweitert:



- Speichern Sie alle Einstellungen in einer Datei vom Typ „rdp“ durch Klicken des Buttons „Save As…“.

© KEBA 2006



Automatischer Start einer RDP-Verbindung via Registry-Eintrag

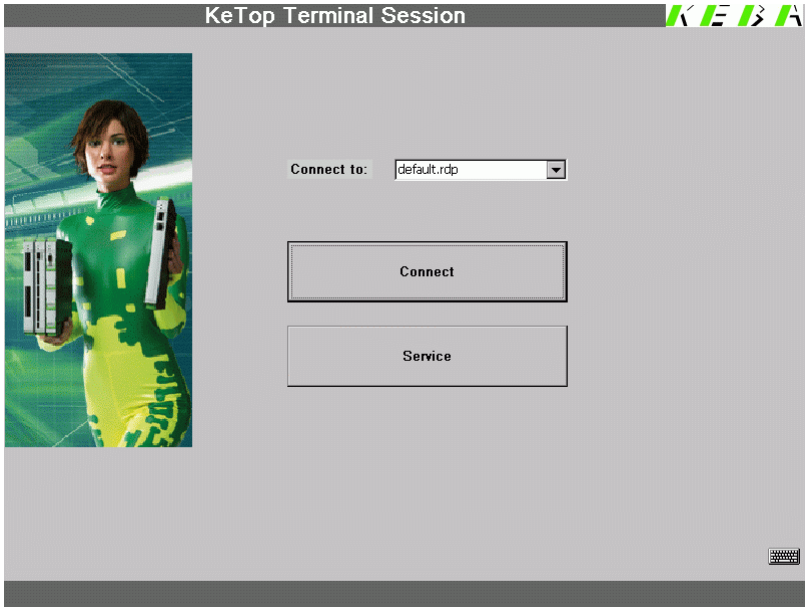
Beim einschalten des KeTop kann eine Verbindung zum Zielrechner automatisch hergestellt werden durch einen Eintrag in der Registry unter [HKEY_LOCAL_MACHINE\Autostart]. (siehe auch Kapitel „Programm zum Starten von Applikation und KeTop API“)

Startx = "\windows\cetsc.exe "
Paramsx = "\ipsm\default.rdp"

Die korrekten RDP-Einstellungen müssen hierbei in der Datei "\ipsm\default.rdp" hinterlegt sein.

Start einer RDP-Verbindung via TSCDialog

TSCDialog.exe ist eine von KEBA erweiterte Version des RDP-Clients. Mit diesem Programm kann der Zugriff des Bedieners auf Windows CE reglementiert werden. Weiters ist eine Gestaltung des Anmeldebilds nach eigenen Wünschen möglich.



© KEBA 2006

Zum automatischen Start von TSCDialog.exe beim Einschalten des KeTop muss folgender Eintrag in der Registry existieren:

HKEY_LOCAL_MACHINE\Autostart\Startx = "\windows\tscdialog.exe"

Die korrekten RDP-Einstellungen müssen hierbei in der Datei "\ipsm\rdpfiles\default.rdp" hinterlegt sein.

Zur Anpassung an Kundenwünsche können einzelne Elemente des Anmeldebilds durch kundenspezifische Elemente ersetzt werden.

Die Anpassung wird geregelt über die Datei
\ipsm\rdpfiles\resources\TSCDialog.ini

TakeUserLogo:i:1

ProgTitle:s:title	(Überschrift)
LogoFileName:s:logo.bmp	(Logo rechts oben, anstelle “KEBA”)
ImgFileName:s:bitmap.bmp	(Bild links, anstelle grüner Frau)

Die Dateien „logo.bmp“ und „bitmap.bmp“ müssen ebenfalls im Verzeichnis „\ipsm\rdpfiles\resources\“ stehen.

© KEBA 2006

KVC – KEBA Virtual Channel

Das Protokoll "KEBA Virtual Channel" (KVC) dient zur Übertragung von Steuer- und Bedienelementdaten zwischen einer Steuerung und einem oder mehreren KeTop-Handbediengeräten.

Folgende Daten können über den KVC übertragen werden:

Daten	Richtung ST ⇔ HT	Werte-bereich	Größe [Bytes]	Übertragungsart
Override-Potentiometer	⇦	0..127	1	Event bei Änderung und auf Anforderung der Steuerung
Elektronisches Handrad	⇔	-32000 .. 32000	2	Event bei Änderung, auf Anforderung der Steuerung und als SetKommando zum Adjustieren
Joystick	⇦	2x –63..63	2	Event bei 0-Pos und != 0-Pos, dann Anforderung von Steuerung
Tasterbeleuchtung f. Taster unter Display	⇔	ein blinken aus	2	Als Kommando von der Steuerung und als Anforderung vom Client zur Steuerung
LEDs am Keyboard				
Kontrast, Helligkeit	⇔	0..255	2	Als Kommando von der Steuerung und als Anforderung vom Client zur Steuerung
Zeit f. Screensaver	⇔	0..255	1	Als Kommando von der Steuerung und als Anforderung vom Client zur Steuerung
Zustand des Screensaver	⇦	0..1	1	Event oder als Anforderung vom Client zur Steuerung
Lautstärke	⇔	0..255	1	Als Kommando von der Steuerung und als Anforderung vom Client zur Steuerung
Hintergrundbeleuchtung	⇔	0..1	1	Als Kommando von der Steuerung und als Anforderung vom Client zur Steuerung
WriteToFlash	⇨	-	-	Kommando
PlaySound	⇨	0..255	1	Kommando
KeepAlive	⇔	0..65535	2	Kommando und KeepAliveEvent wenn vom Server parametriert

Die Datenübertragung zwischen der Steuerung und den Handbediengeräten basiert auf einer Ethernetverbindung (TCP/IP-Protokoll, Listening Port **0xCEBA**), wobei alle Geräte durch ihre IP-Adresse identifiziert werden.

An einer Steuerung können **mehrere** Handbediengeräte angeschlossen werden, jedes Handbediengerät hat aber nur **eine** Verbindung zu **einer** Steuerung. Das KVC-Protokoll ist ein ereignisorientiertes Protokoll, d.h. jeder Teilnehmer kann von sich aus jederzeit Daten senden.

Um die Funktionstüchtigkeit des KeTop T100 in der Steuerung zu überwachen, bietet der KVC die Möglichkeit KeepAlive-Datenpakete zu verwenden, welche periodisch vom Client zum Server gesendet werden. Die Zeitabstände der Datenpakete sind in Millisekundenschritten einstellbar.

KEBA © KEBA 2006

Ereignisse, die den Client zum Senden von Daten veranlassen sind:

- Änderung des Wertes des Override-Potentiometers
- Änderung des Handradwerts
- Änderung des Tastenstatus einer beliebigen Taste
- Erreichen der 0-Position des Joysticks
- Verlassen der 0-Position des Joysticks
- Datenabfragen des Servers

Bei jedem dieser Ereignisse sendet der Client an den Server ein Datenpaket mit dem Eintrag, um welches Event es sich handelt sowie der aktuellen Daten aller Bedienelemente. Werden auf dem Client Ereignisse schneller produziert, als der Server diese verarbeiten kann (z.B. schnelles Drehen des Handrades und gleichzeitiges Bewegen des Joysticks), werden diese Events am Client nicht gesammelt und verzögert gesendet, sondern zusammengefasst.

Ereignisse des Servers

Durch Senden eines Pakets an den Client hat der Server die Möglichkeit, den Status der Anzeigen (Tasten-LEDs) am Client zu beeinflussen, den aktuellen Wert des Handrades zu setzen oder die aktuellen Werte (Positionen von Joystick, Handrad und Overridepotentiometer sowie Zustand der Tasten) abzufragen. Der Server kann auch die aktuellen Statis der durch den Server veränderbaren Werte auslesen.

© KEBA 2006

Datenübertragung

Zwischen Client und Server bilden zwei Socket-Verbindungen jeweils einen Kanal für die Datenübertragung in jede Richtung. Die für die jeweilige Datenrichtung massgeblichen Datenstrukturen werden in diesem Abschnitt beschrieben.

Client → Server

```
typedef enum {
      eKVCJoystickIsZero   0x0001
      eKVCJoystickNotZero  0x0002
      eKVCSpaceMouseIsZero 0x0004
      eKVCSpaceMouseNotZero0x0008
      eKVCHandWheelChanged 0x0010
      eKVCOverrideChanged  0x0020
      eKVCKeyPressed       0x0040
      eKVCKeyReleased      0x0080
      eKVCLedValue         0x0100
      eKVCContrast         0x0200
      eKVCBrightness       0x0300
      eKVCVolume           0x0400
      eKVCScreensaverTime  0x0500
      eKVCScreensaverState 0x0600
      eKVCBacklightState   0x0700
      eKVCLed              0x0800
      eKVCJoystickResp     0x0900
      eKVCSpaceMouseResp   0x0A00
      eKVCHandWheelResp    0x0B00
      eKVCOverrideResp     0x0C00
      eKVCAlive            0xFE00
      eKVCClientDisconnect 0xFF00
} TKVCEvent;

typedef struct {
   UINT16 event;
   struct {
         UINT8                overrideVal;
         UINT8                keyVal;
         TKVCJostickData      joystickVal;
         SINT16               handWheelVal;
         TKVCSpaceMouseData   spaceMouseVal;
   } data;
} TKVCClientData;

typedef enum {
      eKtpKeyboardLedOff   = 1,
      eKtpKeyboardLedOn    = 2,
      eKtpKeyboardLedBlink = 3
} TKtpLedState;

typedef struct {
   char posX;
      char posY;
      char posZ;
```


KEBA © KEBA 2006

```
} TKVCJostickData;

typedef struct {
    UINT16 posX;
    UINT16 posY;
    UINT16 psoZ;
} TKVCSpaceMouseData;

typedef struct {
   SINT16 absVal;
   SINT16 dynVal;
}TKVCHandWheelData;
```

Server → Client

```
typedef enum {
    eKVCSetWheelValue,
    eKVCSetLed
       eKVCSetContrast,
       eKVCSetBrightness,
       eKVCSetVolume,
    eKVCSetScreensaver,

    eKVCGetLed,
    eKVCGetContrast,
    eKVCGetBrightness,
    eKVCGetVolume,
    eKVCGetScreensaverTime,
    eKVCGetJostickValue,
       eKVCGetSpaceMouseValue,
       eKVCGetOverrideValue,
       eKVCGetWheelValue,

       eKVCSwtichBacklight,
       eKVCGetBacklightState,

       eKVCPlaySound,
       eKVCWriteFlash
       eKVCDisconnect
} TKVCCommand;

typedef struct {
    TKVCCommand command;
    SINT16      param;
} TKVCServerData;
```


© KEBA 2006

Beispiel: Interface am Server

Auf der Serverseite wird das KVC-Protokoll durch zwei Klassen repräsentiert: die Klasse `CKVCServer` und `CKVCConnection`. Ein Objekt der Klasse `CKVCServer` repräsentiert den eigentlichen Server (den "Listener") und jeweils ein Objekt der Klasse `CKVCConnection` eine Verbindung zu einem Client.

"Serverklasse"

```
class CKVCServer {
public:
   virtual int Init();
   virtual int Exit();
   virtual CKVCConnection* OnClientConnect(SOCKET socket,
                                           sockaddr_in &sockAdr);
   virtual int OnClientDisconnect(CKVCConnection *pConnection,
                                  TDisconInfo info);
   POSITION ConnectionListHeadPos();
   CKVCConnection* ConnectionListGetNext(POSITION pos);
   int ConnectionListGetCount();

};
```

`int` **`Init`**`();`
Initalisiert den Server aund öffnet den Port 0xCEBA für eingehende Verbindungen.

`int` **`Exit`**`();`
Beendet alle Verbindungen und schließt den Port 0xCEBA.

`CKVCConnection*` **`OnClientConnect`**`(SOCKET socket, sockaddr_in &sockAdr);`
Diese Methode wird immer aufgerufen, wenn ein Handbediengerät eine Verbindung zur Steuerung herstellen möchte. Der Parameter `socket` und `sockAdr` geben die Verbindungsparameter des Handbediengeräts an. Diese Funktion muß einen Zeiger auf ein Objekt der Klasse `CKVCConnection` zurückliefern. Ein Rückgabewert von 0 zeigt an, daß die Steuerung die Anmeldung des Handbediengeräts zurückweist.

`int` **`OnClientDisconnect`**`(CKVCConnection *pConnection, TDisconInfo info);`
Diese Methode wird aufgerufen, wenn der entsprechende Client vom Server nicht mehr erreicht werden kann. Die Ursache des Abmeldens wird in info angegeben.
`POSITION` **`ConnectionListHeadPos`**`();`
Diese Methode liefert die Position des ersten Eintrags in der OpenConnection-Liste.

`CKVCConnection*` **`ConnectionListGetNext`**`(POSITION pos);`
Diese Methode liefert einen Zeiger auf das ConnectionObject gespeichert auf Postiion pos in der der OpenConnection-Liste.

`int` **`ConnectionListGetCount`**`();`
Diese Methode returniert die Anzahl an Verbindungen die in der OpenConnection-Liste gespeichert sind.

KEBA © KEBA 2006

Verbindungen

Die für nachfolgende Funktionen benötigte LED-Nummerierung kann dem Kap. „LED-Nummerierung“ auf Seite 44 entnommen werden.

```
class CKVCConnection {
private:
   char *pIpAdr;
public:
   virtual int Init(CKVCServer *pServer, SOCKET socket,
                    SOCKADDR_IN &socketAdr,Tpriority threadPriority);
   virtual int Exit();
   virtual int OnOverrideChange(SINT16 val);
   virtual int OnWheelChange(SINT16 wheelAbsVal);
   virtual int OnKeyboardEvent(TKVCEvent keyEvent, int keyNum);
   virtual int OnJoystickEvent(TKVCEvent event, TKVCJoystickData *pJData);
   virtual int OnSpaceMouseEvent(TKVCEvent event,
                                    TKVCSpaceMouseData
*pSMData);
   virtual int OnDisconnect(int val);
   virtual int OnAlive(void);

   virtual int GetWheelVal(TKVCHandWheelData &hwData);
   virutal int GetOverrideVal(SINT16 &overrideVal);
   virtual int GetJoystickPos(TKVCJoystickData &jData);
   virtual int GetSpaceMousePos(TKVCSpaceMouseData &smData);
   virtual int GetLedState(UINT8 ledNum,UINT8 &state);
   virtual int GetContrast(UINT8 &contrast);
   virtual int GetBrightness(UINT8 &brightness);
   virtual int GetVolume(UINT8 &volume);
   virutal int GetScreensaverTime(UINT16 &time);
   virtual int GetScreensaverState(UINT8 &state);

   virtual int SetWheelVal(SINT16 val);
   virtual int SetLed(UINT8 ledNum, TKVCLedMode mode);
   virtual int SetContrast(UINT8 contrast);
   virtual int SetBrightness(UINT8 brightness);
   virtual int SetVolume(UINT8 volume);
   virtual int SetScreenSaver(UINT16 screenSaverTime);
   virtual int SwitchBacklight(UINT8 backlightOnOff);

   virtual int WriteToFlash();
   virtual int PlaySound(UINT16 soundNr);
};
Die Methoden OnOverrideChange, OnWheelChange, OnKeyboardEvent,
OnJoystickEvent und OnSpacemouseEvent werden aufgerufen, wenn am
Client beim entsprechenden Bedienelement ein Event aufgetreten ist.

int CKVCConnection::OnOverrideChange(SINT16 val);
Diese Methode wird aufgerufen, wenn sich am Client der Wert des
Override-Potentiometers geändert hat. Der aktuelle Wert wird im
Parameter val angegeben.

int CKVCConnection::OnWheelChange(SINT16 wheelAbsVal);
Die Methode OnWheelChange wird aufgerufen, wenn sich der Wert des
Handrades geändert hat. Der aktuelle Wert wird im Parameter
wheelAbsVal als absoluter Wert übergeben.
```


© KEBA 2006

virtual int **OnKeyboardEvent**(TKVCEvent keyEvent, int keyNum);
Die Methode OnKeyboardEvent wird aufgerufen, wenn eine Taste gedrückt bzw. losgelassen wurde. Die Tastennummer wird in keyNum angegeben, der Zustand der Taste (make, break) in keyEvent.

int CKVCConnection::**OnJoystickEvent**(TKVCEvent event,
TKVCJoystickData *pJData);
Die Methode OnJoystickEvent wird aufgerufen, wenn der Joystick aus der 0-Position bewegt wird bzw. wenn er in die 0-Position kommt. Die aktuelle Position wird im Parameter event und die aktuellen Werte im Parameter pJData mitgeteilt.

int CKVCConnection::**OnSpacemouseEvent**(TKVCEvent event,
TKVCSpaceMouseData *pSMData);
Die Methode OnSpacemouseEvent wird aufgerufen, wenn die Spacemouse aus der 0-Position bewegt wird bzw. wenn sie in die 0-Position kommt. Die aktuelle Position wird im Parameter event und die aktuellen Werte im Parameter pSMData mitgeteilt.

int CKVCConnection::**OnDisconnect**(int val);
Die Methode OnDisconnect wird aufgerufen, wenn der Client die Verbindung per disconnect Meldung beendet.

int CKVCConnection::**OnAliveMsg**();
Die Methode OnAliveMsg() wird aufgerufen, wenn vom Client eine KeepAliveMessage empfangen wurde.

int CKVCConnection::**GetWheelVal**(TKVCHandWheelData &hwData);
Die Methode GetWheelVal liefert den aktuellen Positionswert des Wheels in hwData.absVal und den Änderungswert seit dem letzten Aufruf in hwData.dynVal.

int CKVCConnection::**GetOverrideValue**(SINT16 &overrideVal);
Die Methode GetOverrideValue gibt den aktuellen Wert des Override Potis in der Variable overrideVal zurück.

int CKVCConnection::**GetJoystickPos**(TKVCJoystickData &jData);
Die Methode GetJoystickPos gibt die aktuelle Joystick Position in der Variable jData zurück.

int CKVCConnection::**GetSpaceMousePos**(TKVCSpaceMouseData &smData);
Die Methode GetSpaceMousePos gibt die aktuelle SpaceMouse Position in der Variable smData zurück.

int CKVCConnection::**GetLedState**(UINT8 ledNr , UINT8 &state);
Die Methode GetLedState dient zum Abfragen des aktuellen Status der in ledNr übergebenen Led. Das Ergebnis wird in state zurückgeliefert.

int CKVCConnection::**GetContrast**(UINT8 &contrast);
Die Methode GetContrast gibt den aktuellen Wert der Kontrasteinstellung in der Variable contrast zurück.

int CKVCConnection::**GetBrightness**(UINT8 &brightness);
Die Methode GetBrightness gibt den aktuellen Wert der Helligkeitseinstellung in der Variable brightness zurück.

int CKVCConnection::**GetVolume**(UINT8 &volume);
Die Methode GetVolume gibt den aktuellen Wert der Lautstärkeeinstellung in der Variable volume zurück.

int CKVCConnection::**GetScreensaverTime**(UINT16 &screensaverTime);
Die Methode GetScreensaverTime gibt den aktuellen Wert der Screensavereinstellung in der Variable screensaverTime zurück.

KEBA © KEBA 2006

int CKVCConnection::**GetScreensaverState**(UINT8 &state);
Die Methode GetScreensaverstate gibt den aktuellen Zustand des Screensavers in der Variable state zurück.

int CKVCConnection::**GetBacklightState**(UINT8 &state);
Die Methode GetBacklightState gibt den aktuellen Zustand der Hintergrundbeleuchtung in der Variable state zurück .

int CKVCConnection::**SetWheel**(SINT16 value);
Der Aufruf dieser Methode setzt den Wert des Handrades auf den in value angegebenen Wert und liefert den letzten Wert zurück. Dieser Wert ist der Ausgangswert für den durch OnWheelChange gelieferten Absolutwert.

int CKVCConnection::**SetLed**(UINT8 ledNum, TKVCLedeMode mode);
Der Aufruf der Methode SetLed setzt die Led, die in ledNum definiert ist, auf den in mode übergebenen Modus.

int CKVCConnection::**SetContrast**(UINT8 contrast);
Der Aufruf der Methode SetContrast verändert den Wert der Kontrasteinstellung am Client.

int CKVCConnection::**SetBrightness**(UINT8 brightness);
Der Aufruf der Methode SetBrightness verändert den Wert der Helligkeitseinstellung am Client.

int CKVCConnection::**SetVolume**(UINT8 volume);
Der Aufruf der Methode SetVolume verändert den Wert der Lautstärkeeinstellung am Client.

int CKVCConnection::**SetScreensaver**(UINT16 screensaverTime);
Der Aufruf der Methode SetScreensaver verändert die Ansprechzeit für den Screensaver am Client.

int CKVCConnection::**SwitchBacklight**(UINT8 backlightOnOff);
Der Aufruf der Methode SwitchBacklight schaltet die Hintergrundbeleuchtung am Client ein und aus (backlightOnOff = 1 / backlightOnOff = 0).

int CKVCConnection::**WriteToFlash**();
Der Aufruf der Methode WriteToFlash speichert den Inhalt der Client Windows Registry in den Flashspeicher.

int CKVCConnection::**PlaySound**(UINT16 soundNr);
Der Aufruf der Methode PlaySound startet die Wiedergabe des Soundes mit der in soundNr übergebenen Nummer.

sockaddr_in **GetSocketAdr**();
Die Methode GetSocketAdr returniert die Eigenschaften der Verbindung;

SOCKET **GetSocket**();
Die Methode GetSocket liefert den aktuellen Socket der Verbindung mit dem Client

© KEBA 2006

Server Implementierung

Die Grundlage für den Server sind die beiden Klassen `CKVCServer` und `CKVCConnection`. Die Klasse `CKVCServer` übernimmt den Aufbau und die Verwaltung der Verbindungen. Die Klasse `CKVCConnection` stellt die eigentliche Verbindung dar.
Um einen Server richtig zu implementieren wird eine Ableitung der Klasse `CKVCServer` benötigt.

```
class CKVCTestServer: public CKVCServer{
public:
   CKVCTestServer();
   virtual ~CKVCTestServer();

   CKVCConnection* OnClientConnect(SOCKET socket,
                                   sockaddr_in &sockAdr);
   int OnClientDisconnect(CKVCConnection *pConnection,
                          TKVCDisconInfo info);
};
```

In dieser Klasse muss die Methode `OnClientConnect` überschrieben werden.
In dieser Methode muss ein `CKVCConnection` Objekt angelegt und initialisiert werden.

```
CKVCConnection* CKVCTestServer::OnClientConnect(SOCKET
socket,
 sockaddr_in &sockAdr){
   CKVCTestConnection *pConnect = 0;
   ...
   pConnect = new CKVCTestConnection();
   if (pConnect != 0){
      pConnect->Init(this, socket, sockAdr);
   }
   ...
   return pConnect;
}
```

Die Methode `OnClientDisconnect` kann überschreiben werden, es muss jedoch sichergestellt sein, das in dieser Ableitung die Methode `CKVCServer::OnClientDisconnect` aufgerufen wird.

```
int CKVCTestServer::OnClientDisconnect(CKVCConnection
                                       *pConnection,
                                       TKVCDisconInfo info){
   ...
   return CKVCServer::OnClientDisconnect(pConnection, info);
}
```

In der Ableitung der Klasse `CKVCConnection` müssen nur die Event und Change Methoden abgeleitet werden. Die Get Methoden können abgeleitet werden es muss jedoch sichergestellt werden, das vor Benutzung der Daten die Methode der Basisklasse aufgerufen wird.

```
class CKVCTestConnection: public CKVCConnection{
public:
   CKVCTestConnection();
```


KEBA © KEBA 2006

```
   virtual ~CKVCTestConnection();

   virtual int OnOverrideChange  (SINT16 val);
   virtual int OnWheelChange     (SINT16 wheelAbsVal);
   virtual int OnKeyboardEvent   (TKVCEvent keyEvent, UINT8
                                  keyNum);
   virtual int OnJoystickEvent   (TKVCEvent event,
                                  TKVCJoystickData *pJData);
   virtual int OnSpaceMouseEvent (TKVCEvent event,
                                  TKVCSpaceMouseData
*pSMData);
   virtual int OnAliveMsg        ();
   virtual int OnDisconnect      ();
   virtual int GetWheelVal       (TKVCHandWheelData &hwData);
   virtual int GetOverrideVal    (SINT8 &overrideVal);
   virtual int GetJoystickPos    (TKVCJoystickData &jData);
   virtual int GetSpaceMousePos  (TKVCSpaceMouseData &smData);
   virtual int GetLedState(UINT8 ledNum, UINT8 &state);
   virtual int GetContrast       (UINT8 &contrast);
   virtual int GetBrightness     (UINT8 &brightness);
   virtual int GetVolume         (UINT8 &volume);
   virtual int GetScreensaverTime(UINT16 &screeTime);
   virtual int GetScreensaverState(UINT8 &state);
   virtual int GetBacklightState (UINT8 &state);
};

int CKVCTestConnection::OnOverrideChange  (SINT16 val){
   cout << "OnOverrideChange: " << (int)val << endl << flush;
   return true;
}

int CKVCTestConnection::GetOverrideVal    (SINT8
                                           &overrideVal){
   CKVCConnection::GetOverrideVal(overrideVal);
   cout << "GetOverrideVal: " << (int)overrideVal << endl <<
flush;
   return true;
}
```


© KEBA 2006

Remote-Software ActiveSync

Diese Software dient zum Abgleichen bzw. Übertragen von Daten von und zu einem PC und als Debug-Schnittstelle.

Die ActiveSync Remote-Software ist ein Produkt von Microsoft und kann kostenlos über folgende Downloadadresse bezogen werden:

http://www.microsoft.com/windowsmobile//downloads/activesync38.mspx

Gehen Sie zur Herstellung der Datenverbindung vom KeTop zum PC wie folgt vor:

- ► 1. Entfernen der Anschlussschachtabdeckung am KeTop
- ► 2. Downloadkabel KeTop XD040 an PC und KeTop anstecken
- ► 3. Starten der ActiveSync-Software am PC
  a) bestehende Partnerschaften löschen
  b) ( ) Verbindung durch serielles Kabel... deaktivieren -> <OK>
  c) (x) Verbindung durch serielles Kabel... wieder aktivieren -> <OK>
- ► 4. Starten der ActiveSync-Software am KeTop:
  Start -> Programs -> Communication -> Active Sync
- ► 5. Am PC:
  - KeTop als Gast anmelden (**keine Partnerschaft festlegen**).

Die Datenverbindung zwischen KeTop und PC ist dann aufgebaut. Bei Kommunikationsproblemen beachten Sie noch folgende Hinweise:

- Max. Zeitdifferenz zwischen Schritt 3c) und 4. = 30s.
- Für weitere Versuche sind die Schritte 3. bis 5. zu wiederholen.
- Die ActiveSync-Software am KeTop muss manuell gestartet werden. Ein alleiniges Anstecken des Verbindungskabels bewirkt kein Starten der ActiveSync-Software.
- Ab Werk ist für die Remote-Verbindung eine Baudrate von 115 kBaud vordefiniert. Bei Kommunikationsproblemen durch überlange Leitungen kann durch Verringern der Baudrate Abhilfe geschaffen werden. Die Baudrate kann wie folgt geändert werden:

  - Start -> Programs -> Communication -> Remote Networking
  - "com1_115k" anwählen
  - File -> Properties
  - SP1 on COM1: -> Configure -> Port Setttings
  - Alle Fenster mit <OK> schließen

  (optional: Start -> Programs -> KeTop -> Registry Backup -> <OK>)

KEBA © KEBA 2006

Testtool (VC++ Demo)

Voraussetzungen

Vom Leser werden Kenntnisse in objektorientierte Programmierung, embedded C++ und MFC vorausgesetzt. Die Diagramme sind in UML-Standard nach OMT dargestellt.

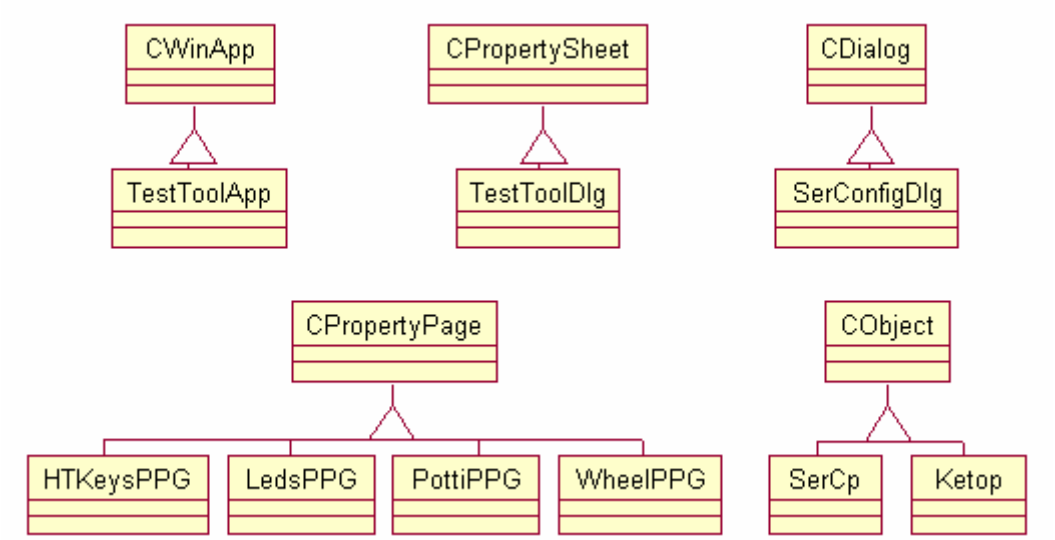
Aufgabe des Testtools

Das Testtool ist eine C++ - Demo und dient zum Anzeigen von Werten am Bildschirm, die vom KetopAPI zur Verfügung gestellt werden, wie z.B. Wheel-Werte, Poti-Werte, Tastenstati und LED-Stati.
Diese Werte werden durch das SER-Protokoll (CP001-Datenprofil) zur Steuerung übertragen, von dort sofort wieder ausgelesen und ebenfalls am Bildschirm dargestellt.

Realisiert wurde dieses Testtool in embedded Visual C++ 3.0® unter Verwendung der MFC® (für WinCe).

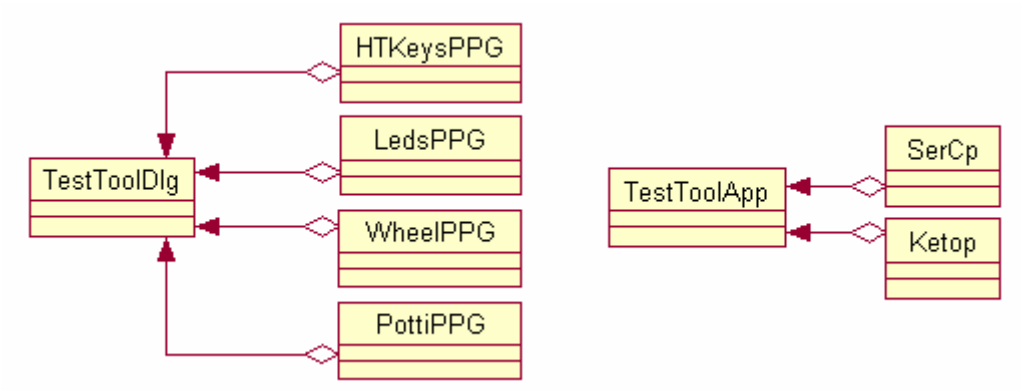
Beschreibung der Klassen

Klassenhierarchie



© KEBA 2006

Klassenbeziehungen



TestToolApp

Diese Klasse ist die eigentliche Applikation, die am Bildschirm dargestellt wird. In ihrer Init-Methode legt sie den Dialog (repräsentiert durch die Klasse TestToolDlg) an und verwaltet ihn. Da während der Laufzeit des Programmes auf einfache Weise auf die Applikation zugegriffen werden kann, enthält diese jeweils eine Membervariable auf die beiden Abstraktionsschichten SerCp und Ketop.

TestToolDlg

Diese Dialogklasse enthält die einzelnen Tabreiter, die durch die Property-Page-Klassen realisiert sind.

HTKeysPPG

Diese Klasse repräsentiert den Tabreiter HTKeys und kümmert sich um das Visualisieren von Tastendrücke.

LedsPPG

Diese Klasse ist für die Ledsverwaltung zuständig und wird als Leds-Tabreiter dargestellt.

PottiPPG

Diese Klasse kümmert sich um alle Werte des Potentiometers und ist durch den Potentiometer-Tabreiter visualisiert.

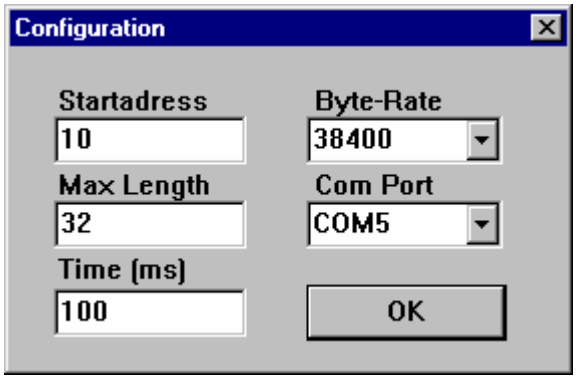
WheelPPG

Diese Klasse stellt alle Veränderungen am Wheel-Rad im Tabreiter Wheel dar.

SerConfigDlg

Diese Klasse implementiert die Funktionalität des "Konfigurationsdialoges" (siehe folgende Abbildung), der beim Starten der Applikation am Bildschirm erscheint. Dadurch hat der Benutzer die Möglichkeit, eigene Portkonfigurationen einzugeben.

KEBA © KEBA 2006



Da die Werte dieses Dialoges bei der Initialisierung des Ports schon zur Verfügung stehen müssen, wird er von der Klasse SerCp gehalten, da er hier designmäßig am besten hineinpaßt.

SerCp

Diese Klasse bildet eine Abstraktionsschicht über den CP001-Treiber und soll die Kommunikation mit der Steuerung für die Applikation darüber vereinfachen.

In ihrem Constructor erfolgt das Laden der SerCp001.dll. Weiters werden sofort Funktionszeiger auf die verwendeten Methoden der SerCp001.dll angelegt und die Callback-Funktion, die zum Auslesen von Werten auf der Steuerung dient, eingehängt. Der Destructor kümmert sich um die Schließung des Ports und um die Freigabe der Dll. Diese Klasse bietet eine "vereinfachte" Schnittstelle zur Kommunikation mit der Steuerung an.

Ketop

Diese Klasse bildet ebenfalls eine Abstraktionsschicht über die KetopAPI und soll so die Zugriffe darauf vereinfachen. Sie bietet nach außen etwas "vereinfachtere" Schnittstellen an.

In ihrem Constructor wird die KetopAPI.dll geladen. Auch hier werden Funktionszeiger auf die benötigten Methoden geholt (jedoch auf eine etwas andere Art und Weise, als in der Klasse SerCp) und eine Callback-Funktion eingehängt, die von der KetopAPI automatisch bei einer Änderung aufgerufen wird.

Vorgehensweise am Beispiel des Testtools

Im Init der TestToolApp-Klasse wird zuerst die Membervariable (mSerCp) der Klasse SerCp, die sich um die Konfiguration des Treibers kümmert, initialisiert. Dabei wird der Konfigurationsdialog (SerConfigDlg) aufgerufen, die eingegebenen Werte abgespeichert und mit ihnen der Port zur Steuerung geöffnet und konfiguriert. Danach wird (in der Klasse TestToolApp) der TestToolDialog aufgerufen und am Bildschirm dargestellt. Dieser startet schon zu Beginn einen Timer, der zyklisch einen Write-Befehl mit dem Lesekommando zur Steuerung absetzt. (Ohne Timer wird die Applikation nie

© KEBA 2006

über Änderungen auf der Steuerung benachrichtigt, da bei diesem Testbeispiel die Steuerung selbst nie von allein aktiv wird.)

Wird nun z.B. am Wheel-Rad gedreht, wird automatisch die Callback-Funktion der KetopAPI aufgerufen. Dort wird eine Windows-Message verschickt und von der Testtool-Applikation abgefangen und abgearbeitet, was in diesem Fall gleichbedeutend mit einem graphischen Update des gerade aktiven Tabreiters ist. Danach wird dieser veränderte Wert durch einen Write-Befehl zur Steuerung geschickt.

Der gestartete Timer teilt der Steuerung zyklisch mit, daß er auslesen möchte. Die Steuerung ruft die bei ihr registrierte Callback-Funktion auf und löst dadurch eine Window-Message aus. Diese wird wiederum von der Applikation abgefangen. Als nächster Schritt wird ein Lesebefehl zur Steuerung geschickt und der gerade aktive Tabreiter ermittelt. Diesem werden die geänderten Werte übergeben und nur er führt eine Aktualisierung am Bildschirm durch.

Allgemeine Vorgehensweise zur Kommunikation mit der Steuerung unter Verwendung der SerCp001.dll

1. Library (SerCp001.dll) laden
2. Funktionszeiger auf die benötigten Methoden holen
3. Callback-Funktion installieren
4. Port öffnen und konfigurieren
5. Verwendung der Methoden, die von der Dll zur Verfügung gestellt werden.
6. Port wieder schließen
7. Library freigeben

KEBA © KEBA 2006

8 Bedienhinweise für das KeTop

Dieses Kapitel beschreibt Abweichungen von herkömmlichen Windows CE-Geräten und KeTop-spezifische Einstellungen.

Hinweis

- *Uhrzeit und Datum werden im KeTop nicht gepuffert und müssen daher, sofern sie für den Benutzer relevant sind, nach dem Aus-/Einschalten wieder neu gesetzt werden. Datum und Uhrzeit können zB bei den Protokolldaten sehr wichtig sein.*

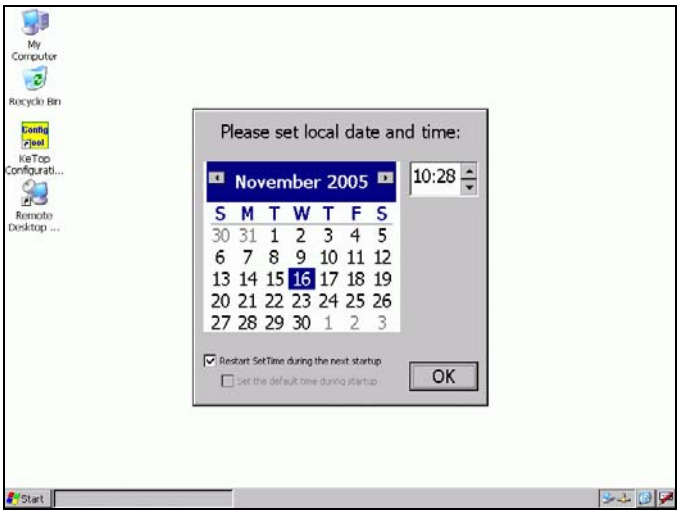
Hinweis

- *Bei Einstellungen, die nicht mit dem ConfigTool durchgeführt werden, müssen die Veränderungen wie folgt abgespeichert werden:*

  Start -> Programs -> KeTop -> Registry Backup
  (oder im Explorer mit dem Kommando „\windows\RegistryBackup.exe“)

Einstellen von Datum und Uhrzeit

Das Zeit- und Datum-Eingabefenster wird standardmäßig bei jedem neuen Hochlauf des KeTops angezeigt.
Da im KeTop keine Echtzeituhr vorhanden ist, ist bei Verwendung der Uhrzeit diese bei jedem Hochlauf neu einzugeben.


My Computer
Recycle Bin
KeTop Configurati...
Remote Desktop ...
Please set local date and time:
November 2005
10:28
S M T W T F S
30 31 1 2 3 4 5
6 7 8 9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 1 2 3
Restart SetTime during the next startup
Set the default time during startup
OK
Start

Der Aufruf kann unterdrückt werden, indem man die obere Checkbox "Restart Settime during the next startup" deaktiviert.
In diesem Fall haben Datum und Uhrzeit Zufallswerte.
Wird hingegen die untere Checkbox "Set the default time during startup" aktiviert, dann werden Erstelldatum und -Uhrzeit herangezogen.
In beiden Fällen muss die Änderung der Einstellung per Registry Backup gesichert werden. ("Start -> Programs -> KeTop -> Registry Backup").

© KEBA 2006

KeTop Configuration Tool (ConfigTool)

Das ConfigTool dient zur Kalibrierung und zur einfachen Funktionskontrolle der Bedienelemente, zur Auswahl der Ethernet-Schnittstelle und zur Einstellung von Startup-Funktionen.

Es kann im Start-Menü wie folgt aufgerufen werden:

Start -> Programs -> KeTop -> KeTop Configuration Tool
(oder durch Doppelklicken des Symbols „ConfigTool“ im Windows-Desktop)

Hinweis

- *Das Erscheiniungsbild des Config Tools ist von den im KeTop installierten Bedienelementen abhängig (die Menüs fuer OV-Potentiometer, Joystick und Handrad werden nur angezeigt wenn die Bedienelemente auch im Gerät eingebaut sind).*

Nachdem Veränderungen an der Konfiguration vorgenommen wurden und das Programm beendet wird, hat man die Möglichkeit, sämtliche Änderungen per "Registry Backup" resistent in der Registry zu speichern.

Bei bestimmten Änderungen (zB Kalibrierung) erscheint beim Beenden des Programmes ein Fenster mit folgender Meldung und Auswahlmöglichkeit:

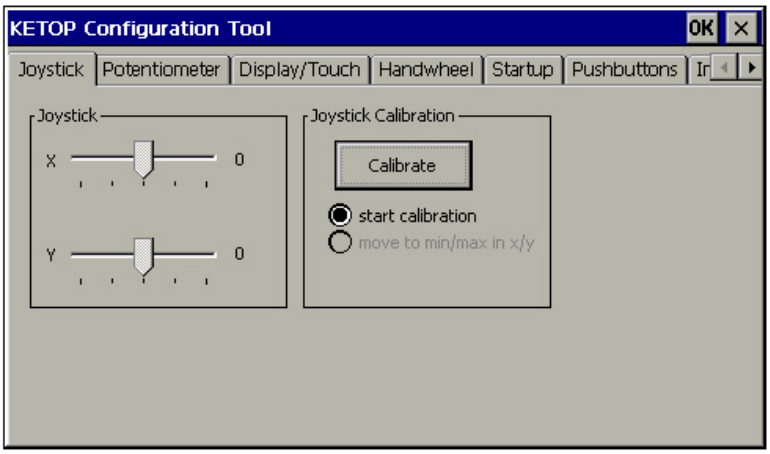
Data has changed! Do you want to write the registry to the flash? Yes/No

Bei der Auswahl „*No*“ werden die Einstellungen nicht resistent im DRAM gespeichert und gehen beim Abschalten des KeTops wieder verloren.

Bei der Auswahl „*Yes*“ wird die Kalibrierung resistent im Flash gespeichert und bleibt auch beim erneuten Einschalten des Gerätes erhalten. Das Gerät ist während des Speichervorganges (dauert max. 10 s) blockiert.

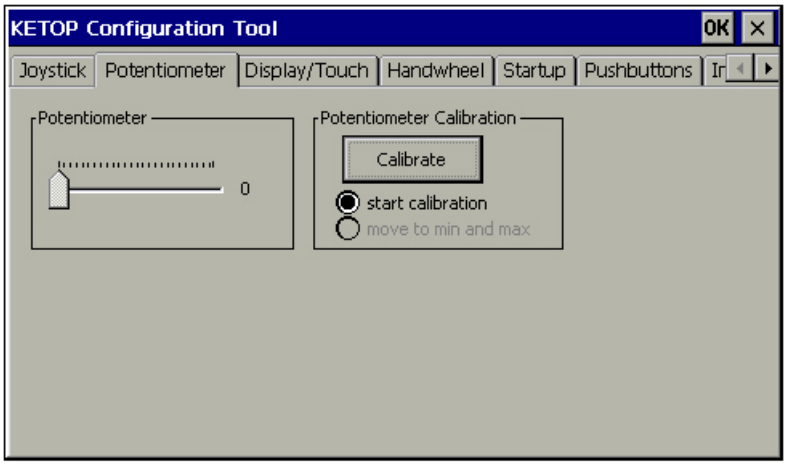
KEBA © KEBA 2006

Kalibrierung Joystick



Neben dem Anzeigebalken wird eine Zahl angezeigt: Während des Kalibriervorganges gibt diese Zahl den aktuellen Wert des ADC (Wertebereich: 0..1024) an und hat nur den Sinn einer Funktionskontrolle. Nachdem die Kalibrierung abgeschlossen ist, gibt die Zahl den aktuellen Wert des Potentiometers an (Wertebereich: 0..127).
Dieser Wert ist auch der Wert, den die Applikation durch die KeTop-API-Funktion *KtpGetOverridePoti* geliefert bekommt.

Kalibrierung Potentiometer

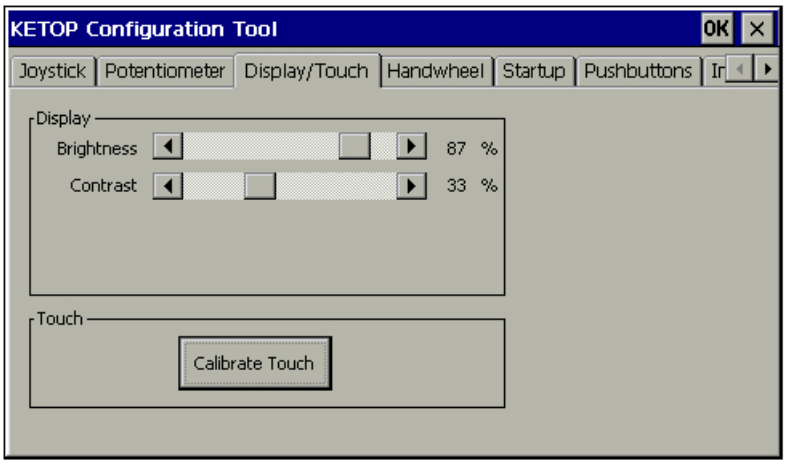


Neben dem Anzeigebalken wird eine Zahl angezeigt: Während des Kalibriervorganges gibt diese Zahl den aktuellen Wert des ADC (Wertebereich: 0..1024) an und hat nur den Sinn einer Funktionskontrolle. Nachdem die Kalibrierung abgeschlossen ist, gibt die Zahl den aktuellen Wert des Potentiometers an (Wertebereich: 0..127).
Dieser Wert ist auch der Wert, den die Applikation durch die KeTop-API-Funktion *KtpGetOverridePoti* geliefert bekommt.

© KEBA 2006

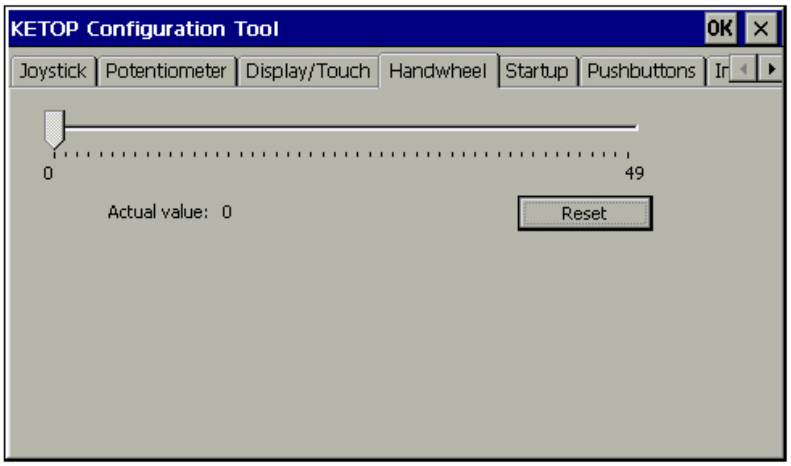
Display- und Touchscreeneinstellungen

Hier erfolgen für das Display die Einstellungen von Helligkeit und Kontrast.



Desweiteren kann der Kalibriervorgang für den Touch-Screen durchgeführt werden.

Nullpunktsetzen des Handrades

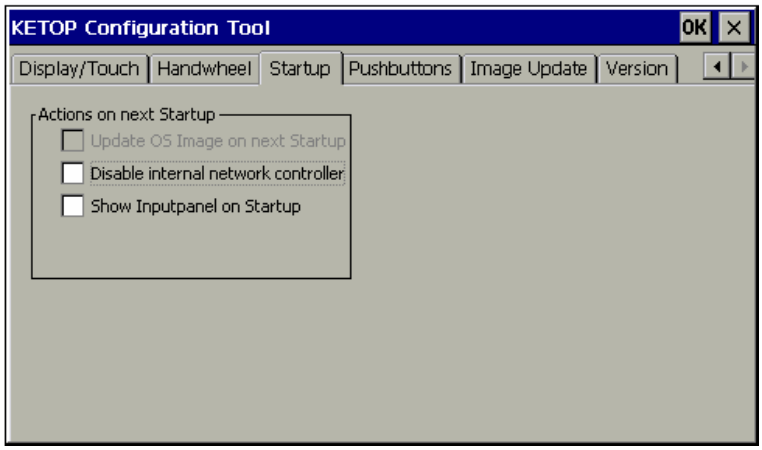


Das Handrad liefert einen 16bit-Wert, der in der Kundenapplikation beliebig verarbeitet werden kann (z. B.: 0-65535, +/-32767, ...).
In der Dialogbox wird der Wert von 0 bis 65535 angezeigt. Der Wert des Handrades wird beim Einschalten des Gerätes auf 0 gesetzt.
Bei Betätigung der Taste „Reset“ wir der Wert des Handrades auf 0 zurückgesetzt.

KEBA © KEBA 2006

Einschaltaktionen für Neustart (Startup)

Diese Toolbox ermöglicht die Festlegung von Einschaltaktionen nach einem Neustart des KeTops.



- *Update OS Image on next Startup (derzeit inaktiv und nicht anwählbar):*
  Durch Auswahl dieses Punktes wird beim nächsten Hochfahren des KeTops automatisch ein OS(Operating System)-Image vom BOOTP-Server über das Netzwerk geladen.

Hinweis
Voraussetzung hierfür ist eine Ethernetverbindung, ein richtig konfigurierter BOOTP-Server sowie eine Image-Datei auf dem Server.

- *Disable internal network controller:*
  Durch Auswahl dieses Punktes wird beim nächsten Hochfahren die interne Ethernet-Schnittstelle „CELAN1: Onboard Ethernet“
  (vgl. *Start -> Settings -> Control Panel -> Network -> Network Configuration*) ausgeschaltet.

Hinweis
Diese Funktion wird benötigt wenn zB eine Ethernet-Schnittstelle im PCMCIA Schacht installiert werden soll.

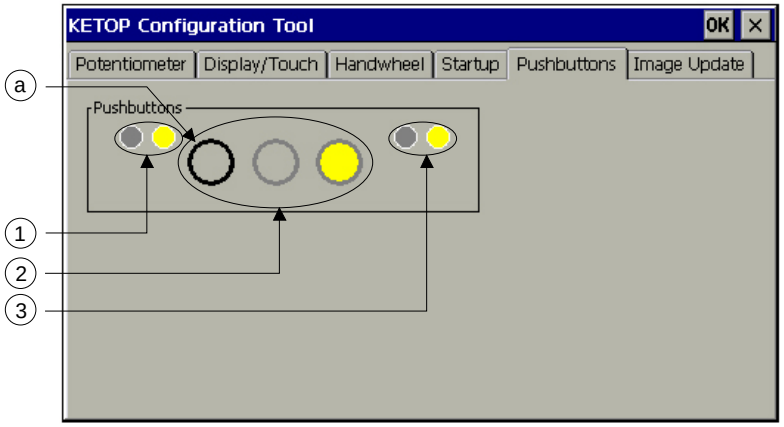
Die eigentliche Konfiguration der Ethernet-Controller muss in der Systemsteuerung erfolgen.

- *Show Inputpanel on Startup:*
  Durch Auswahl dieses Punktes wird bei jedem Start von Windows CE das Softkeyboard eingeblendet.

© KEBA 2006

Tasterkontrolle (Pushbuttons)

Diese Toolbox ermöglicht eine einfache Funktionskontrolle der Taster des KeTops.



1..... Symbole für die linken LEDs der Folientastatur (wenn vorhanden)

2..... Symbole für Bedienelemente

3..... Symbole für die rechten LEDs der Folientastatur (wenn vorhanden)

a..... Schwarzer Ring signalisiert Betätigung eines Bedienelementes

Durch Klicken auf eine der drei Symbole für Bedienelemente (2) können die entsprechenden LED's der Bedienelemente für eine einfache Funktions-kontrolle ein, aus, oder blinkend geschaltet werden. Nach dem ersten Klick wird die Lampe auf blinkend geschaltet, der folgende Klick schaltet die Lampe ein und der nächste Klick wieder aus.

Gleiches gilt für die LEDs der Folientastatur, falls vorhanden (1 und 3).

Die Betätigung eines Bedienelementes wird durch einen schwarzen Ring (ansonsten grau) signalisiert (a).

KEBA © KEBA 2006

Laden einer Image-Datei

Diese Toolbox dient zum Laden einer Image-Datei in das KeTop.

Hinweis

- *Die Image-Datei darf in jedes beliebige Verzeichnis, nur nicht in das Verzeichnis \IPSM kopiert werden.*
- *Vor jedem Image-Update unbedingt „Check File“ ausführen.*

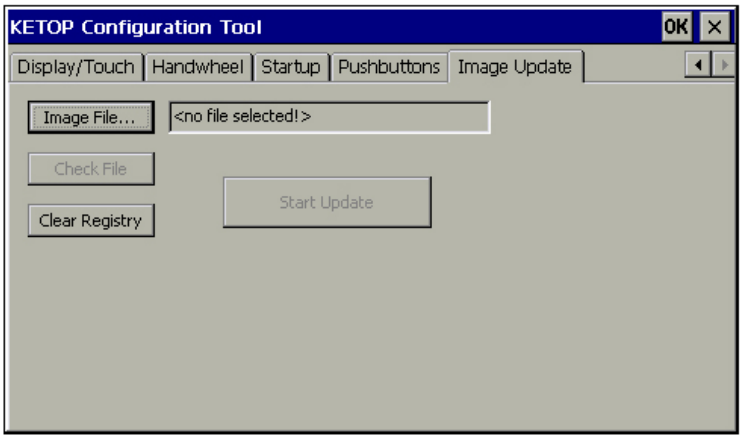


Image File Durch Klicken auf diesen Button kann ein neues WinCE-Imagefile im Dateiauswahldialog gewählt werden.

Hinweis

Im Falle eines Image updates wird automatisch die Registry gelöscht und das KeTop rebootet. Das KeTop fährt nachfolgend mit den Fabrikseinstellungen hoch und alle Einstellungen (Touchscreen, Joystick, Override-Potentiometer, Systemsteuerung,...) müssen neu gespeichert werden.

Check File Das ausgewählte File kann auf Gültigkeit überprüft werden. "Check File" ist nur aktiv, wenn ein gültiges Imagefile ausgewählt wurde.

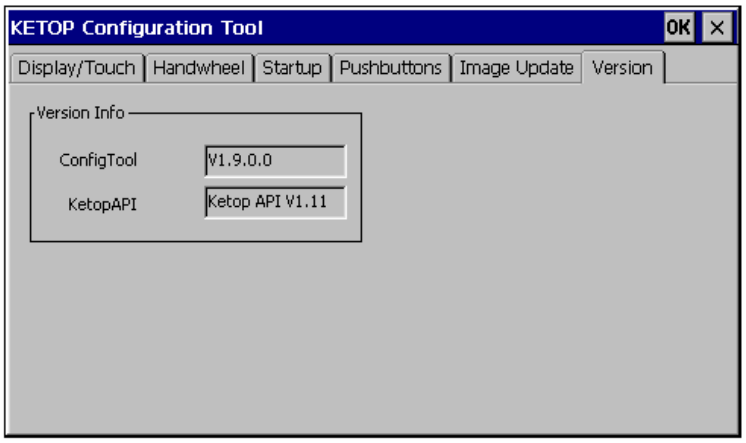
Start Update Mit diesem Button wird der Updatevorgang gestartet und der Fortschrittsstatus im Progressbar darunter angezeigt. "Start Update" ist nur aktiv, wenn das ausgewählte Imagefile (durch "Check File") erfolgreich auf Gültigkeit geprüft wurde.

Clear Registry Durch Auswahl dieses Punktes wird die Registry gelöscht und das KeTop automatisch rebootet. Das KeTop fährt nachfolgend mit den Fabrikseinstellungen hoch und alle Einstellungen (Touchscreen, Joystick und Override-Potentiometer, Systemsteuerung,...) müssen neu gespeichert werden.

© KEBA 2006

Version Info

Diese Fenster gibt Auskunft über die am KeTop installierte ConfigTool- und KetopAPI-Version.



KEBA © KEBA 2006

Überprüfen der Bedien- und Steuerelemente

Um die Funktionalität der Bedien- und Steuerelemente des KeTops zu überprüfen, steht am KeTop die Software „Systemcheck“ zur Verfügung. Diese Software ist intuitiv bedienbar und befindet sich im Startmenü. Sie wird wie folgt aufgerufen:

Start -> Programs -> KeTop -> „System Check“

Folgende Bedien- und Steuerelemente können, soferne sie am Gerät vorhanden sind, überprüft werden:

- Override Potentiometer
- Elektronisches Handrad
- Joystick
- Status-LEDs
- Folientastatur
- Touch-Screen
- Display
- Leuchtdrucktaster
- Schlüsselschalter
- Wahlschalter
- PC-Card Schnittstelle (Slot), falls vorhanden.

Folgende Steuerelemente können NICHT überprüft werden, da eine interne Auswertung über die Elektronik nicht vorgesehen ist:

- Not-Aus-Taster
- Zustimmungstaster

© KEBA 2006

Installation von Programmen

Programme können nachträglich nur in dem Verzeichnis IPSM (Intel Persistent Storage Manager) installiert werden. Da nur dieses Verzeichnis resistent im Flash gespeichert wird, gehen anderweitig installierte Daten nach dem Ausschalten des KeTops verloren.

Eventuelle Installationseinträge in der Registry können mittels dem Kommando „\Windwos\Regflush.exe“ gespeichert werden.

Hinweis

- *Systemdateien, die bei der Installation in das \windows Verzeichnis kopiert werden, bleiben nicht dauerhaft gespeichert. Aus diesem Grund müssen diese Dateien manuell in das Verzeichnis \IPSM\windows (ist gegebenenfalls vom Benutzer anzulegen) kopiert werden. Bei einem Neustart von Windows CE werden diese Dateien automatisch in das \windows Verzeichnis kopiert. Diese Dateien stehen somit dem Betriebssystem wieder zur Verfügung und sind auch nach einem Neustart wieder vorhanden (KeTop T100: ab OEM Build Vers. 1.4a; KeTop T50-ADP: ab OEM Build Vers. 1.2).*

Speichern von Dateien

Das Speichern von Dateien muss im Verzeichnis IPSM erfolgen. Der Inhalt anderer Verzeichnisse wird beim Abschalten nicht gesichert.

Übertragung von Dateien

Übertragung per externem Speichermedium (PC-Card, CF-Card, USB-Stick)

Hinweise zu geeigneten Speichermedien siehe Kapitel "Anschluss", Abschnitt "PC-Card Slot für PC-Cards I, II, III" bzw. "Compact Flash - Card Slot" bzw. „USB-Stick“

- ▶ Stecken Sie das Speichermedium in einen geeigneten PC.
- ▶ Kopieren Sie die gewünschten Dateien auf das Speichermedium.
- ▶ Stecken Sie das Speichermedium in das KeTop.
- ▶ Kopieren Sie die gewünschen Dateien von "\storage card" bzw. " \hard disk“nach "\ipsm".

KEBA © KEBA 2006

Übertragung per Netzwerkverbindung via SMB-Zugriff vom PC auf das KeTop

Diese Art und Weise des SMB-Zugriffs funktioniert nur mit Windows XP Professional und manchen Versionen von Windows 2000.

Allgemeine Hinweise siehe Kapitel "Anschluss", Abschnitt "Ethernet".

Das KeTop muss eine gültige Ethernet-Adresse haben.

Falls sich kein DHCP-Server im Netzwerk befindet, kann die IP-Adresse eingestellt werden über "Start ➔ Settings ➔ Control Panel ➔ Network and Dial-up Connections ➔ CELAN1... ➔ Properties..."

Zum Speichern der IP-Adresse "Start ➔ Programs ➔ KETOP ➔ Registry Backup" aufrufen und anschließend das KeTop neu starten.
(Die Ethernet-Verbindung kann mit "Ping" überprüft werden.)

In der Adresszeile des Windows Explorer am PC kann nun die IP-Adresse des KeTop eingegeben werden (zB: \\192.168.30.107).

Daraufhin hat der PC vollen Zugriff auf die Verzeichnis-Struktur am KeTop und die gewünschten Dateien können vom PC nach "\ipsm" am KeTop kopiert werden.

Übertragung per Netzwerkverbindung via Zugriff vom KeTop auf den PC

Allgemeine Hinweise siehe Kapitel "Anschluss", Abschnitt "Ethernet".

Das KeTop muss eine gültige Ethernet-Adresse haben.

Falls sich kein DHCP-Server im Netzwerk befindet, kann die IP-Adresse eingestellt werden über "Start ➔ Settings ➔ Control Panel ➔ Network and Dial-up Connections ➔ CELAN1... ➔ Properties..."

Zum Speichern der IP-Adresse "Start ➔ Programs ➔ KETOP ➔ Registry Backup" aufrufen und anschließend das KeTop neu starten.
(Die Ethernet-Verbindung kann mit "Ping" überprüft werden.)

Nach dem Hochlauf des KeTop den Windows Explorer am KeTop starten.
(Start ➔ Programs ➔ Windows Explorer)

In der Adresszeile des Windows Explorer am KeTop kann nun das freigegebene Verzeichnis des PC adressiert werden in der Form "\\<pcname>\<freigabename>". (zB: \\tkg007\temp)

Das Verzeichnis am PC muss freigegeben sein und es muss einen User mit den entsprechenden Zugriffsrechten geben.

Nun können die Dateien vom PC nach "\ipsm" am KeTop kopiert werden.

© KEBA 2006

Übertragen via ActiveSync

Hinweise hierzu siehe Kap. „Remote-Software ActiveSync“.

© KEBA 2006

9 Optionen

Dieses Kapitel beschreibt die verschiedenen Optionsmöglichkeiten, mit welchen das KeTop erhältlich ist.

Override Potentiometer

Ist das *KeTop* mit einem Override-Potentiometer ausgestattet, so wird dieses im KeTop softwaremäßig ausgewertet und kann mit einem Programm über das KeTop-API ausgelesen werden.

Das Override-Potentiometer bietet sich für verschiedenste Einsatzmöglichkeiten an, zB zur Einstellung der Spindeldrehzahl und des Vorschubs an Werkzeugmaschinen.

- Auflösung: 0 – 127, linear

Elektronisches Handrad

Ist das KeTop mit einem elektronischen Handrad ausgestattet, werden die Handrad-Impulse im Prozessor ausgewertet und können mit einem Programm über das KeTop-API ausgelesen werden.

Pro Umdrehung werden 50 Impulse gezählt. Eine Drehung des Handrades im Uhrzeigersinn inkrementiert, eine Drehung gegen den Uhrzeigersinn dekrementiert den Zählerstand (-32768 bis +32767, 16 Bit-Wert).

Wesentliche Merkmale

- 50 Rastungen / Umdrehung
- 1 Impuls / Rastung

Hinweis

- *Sollte das KeTop zu Boden fallen, überprüfen Sie den mechanischen Sitz des Drehknopfes. Der Drehknopf kann durch mittiges Andrücken von oben gegebenenfalls wieder eingerastet werden.*

© KEBA 2006

Leuchtdrucktaster

Ist das KeTop mit einem Leuchtdrucktaster ausgestattet, so wird dieser im KeTop softwaremäßig ausgewertet und kann mit einem Programm über das KeTop-API ausgelesen werden.
Die Leuchtdrucktaster sind in „tastender“ (mit Beschriftungsplättchen „0“ oder „I“) oder in „rastender“ (mit Beschriftungsplättchen I/0) Ausführung erhältlich.

⚠ WARNUNG

- **Nach Abnahme der Sichtkappen (zB bei Änderungen an den Beschriftungsplättchen) ist darauf zu achten, dass die Beschriftungsplättchen den Tasterfunktionen entsprechend zugehörig wieder eingesetzt werden.**
  (Es besteht ansonsten Verwechslungsgefahr beim zB Ein- /Ausschalten eines Motors).

Schlüsselschalter

Ist das KeTop mit einem Schlüsselschalter ausgestattet, so wird dieser im KeTop softwaremäßig ausgewertet und kann mit einem Programm über das KeTop-API ausgelesen werden.
Der Schlüsselschalter ist mit 3-Stellungen und der Aufschrift „I-0-II“ erhältlich.

Wahlschalter

Ist das KeTop mit einem Wahlschalter ausgestattet, so wird dieser im KeTop softwaremäßig ausgewertet und kann mit einem Programm über das KeTop-API ausgelesen werden.
Der Wahlschalter ist mit 3-Stellungen und der Aufschrift „I-0-II“ erhältlich.

Joystick

Ist das KeTop mit einem 2-Achsen-Joystick ausgestattet, so wird dieser im KeTop softwaremäßig ausgewertet und kann mit einem Programm über das KeTop-API ausgelesen werden.
Der Knüppel des Joysticks ist aus Fallschutzgründen kurz ausgeführt. Der Joystick ermöglicht zB das Verfahren von Roboterachsen.
Wertebereich: -15 bis +15 je Achse (31 Inkremente)

Der als Option erhältliche 3-Achs-Joystick kann aus platzgründen nicht gleichzeitig mit einem Flashcard-Slot bestückt werden.

KEBA © KEBA 2006

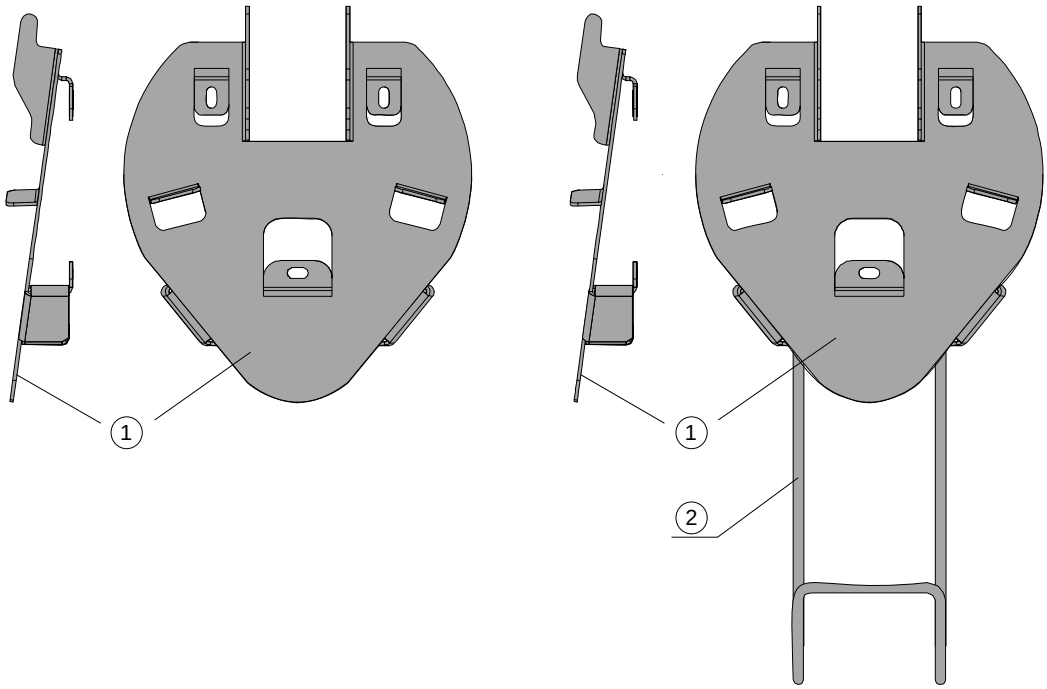
10Zubehör

Wandhalterung KeTop WB090 und KeTop WB095

Die pulverbeschichtete, schwarze Wandhalterung dient zum stationären Betrieb oder zur Ablage des KeTop.

Es wird zwischen 2 Wandhalterungen unterschieden:

KeTop WB090 Wandhalterung ohne Kabelaufhängung	KeTop WB095 Wandhalterung mit Kabelaufhängung

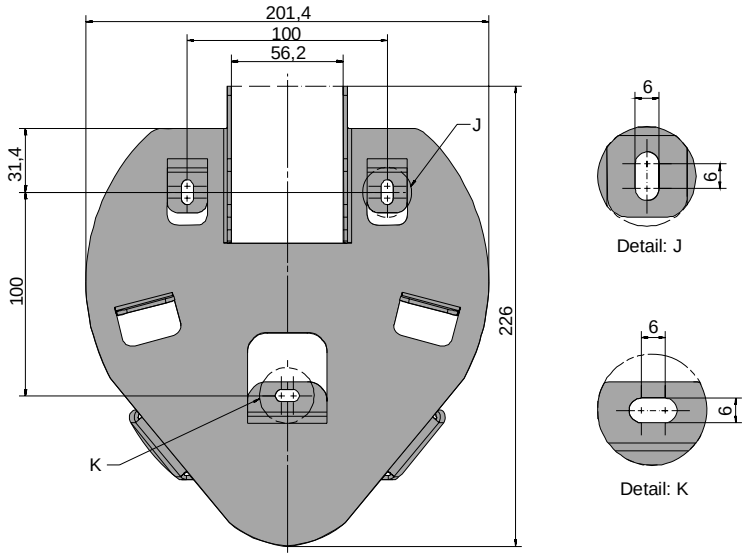


1.....Geräteträger
2.....Kabelaufhängung

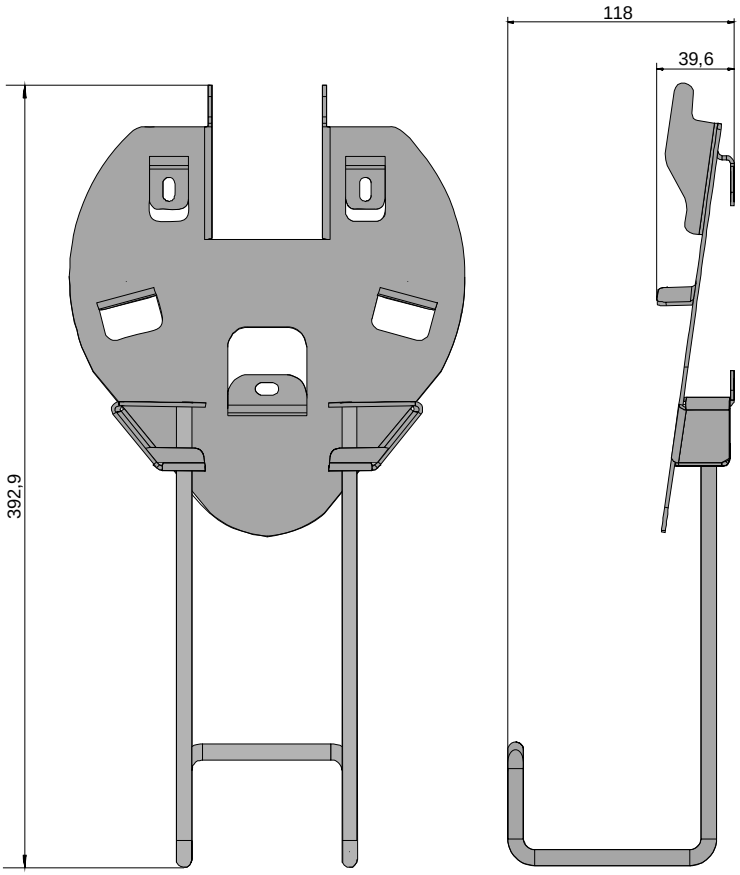
Wandhalterungen KeTop WB090 und KeTop WB095

© KEBA 2006

Maßzeichnungen



Frontansicht Wandhalterung KeTop WB090



Rück- und Seitenansicht der Wandhalterung KeTop WB095

KEBA © KEBA 2006

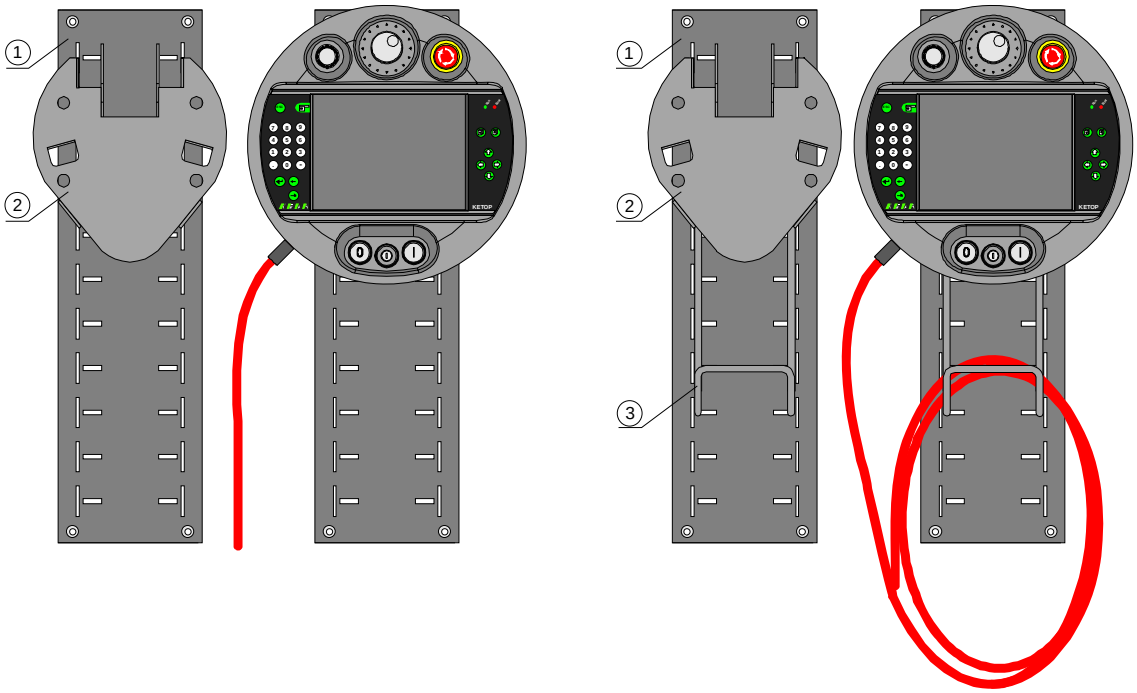
Wandhalterung mit Höhenverstellung KeTop WB100 / WB110

Die pulverbeschichtete, schwarze Wandhalterung dient zum stationären Betrieb oder zur Ablage des KeTop.

Der Geräteträger ist auf der Höhenverstellung über eine Höhe von 320 mm in 8 Stufen verstellbar. Es muss dabei darauf geachtet werden, dass der Geräteträger an allen 4 Punkten in der Höhenverstellung eingehängt wird. Die Kabelaufhängung muss mit den mitgelieferten Schrauben am Geräteträger montiert werden.

Es wird zwischen 2 Wandhalterungen unterschieden:

KeTop WB100 Wandhalterung ohne Kabelaufhängung	**KeTop WB110** Wandhalterung mit Kabelaufhängung



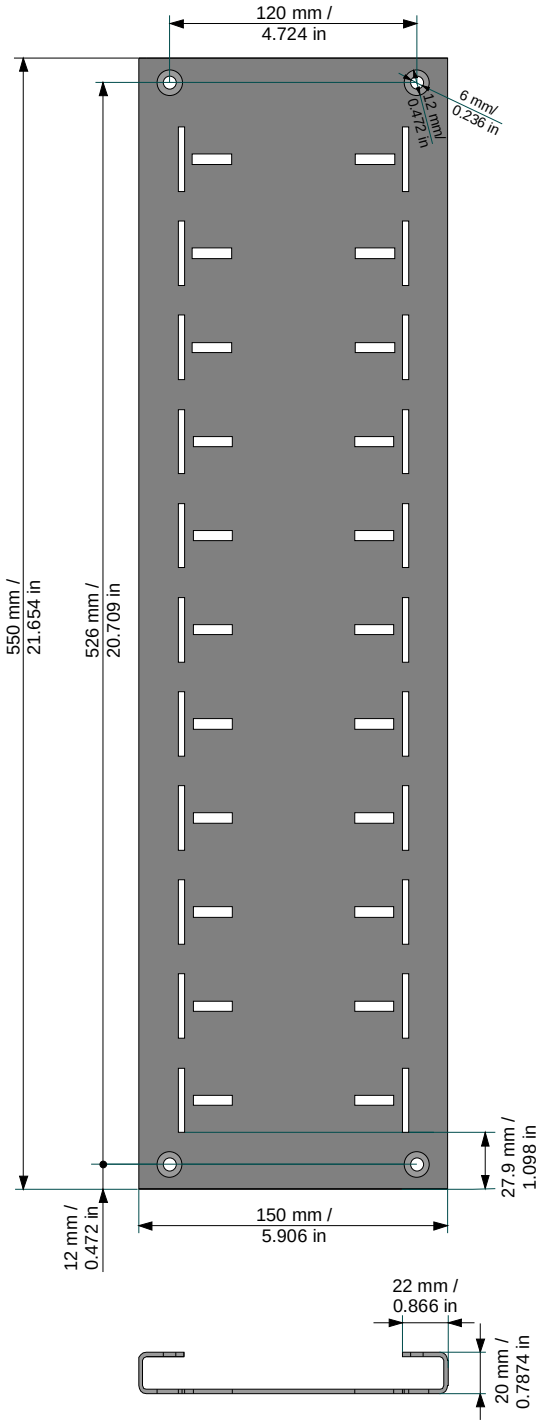
1.....Höhenverstellung
2.....Geräteträger
3.....Kabelaufhängung

Wandhalterungen KeTop WB100 und KeTop WB110 mit und ohne KeTop

© KEBA 2006

Höhenverstellung

Verwenden Sie zur Montage der Höhenverstellung passende Schrauben (sind nicht Bestandteil der Lieferung).



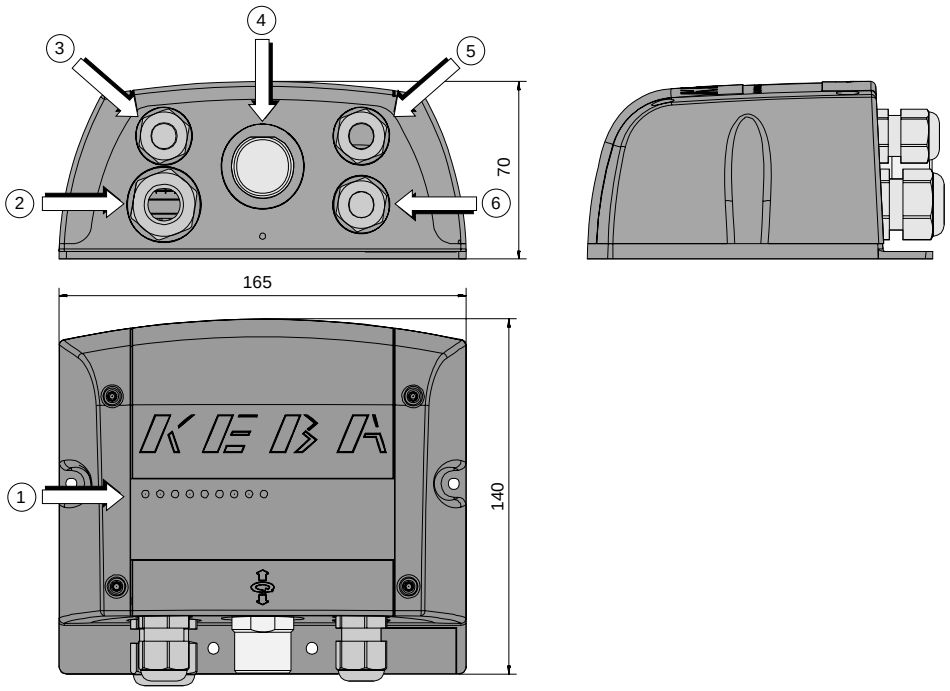
Höhenverstellung der Wandhalterung WB110

KEBA © KEBA 2006

Anschlussbox KeTop CB211

Die Anschlussbox KeTop CB211 dient zur Integration des KeTops in die Maschine/Anlage. Sie ist zur Wandmontage geeignet und kann unter Verwendung des Hutschienenmontagesatzes (KeTop DR200) auch auf einer Hutschiene montiert werden. Die Anschlussbox KeTop CB211 verfügt über folgende Anschlüsse:

Aufbau



1.....Status- und Fehler-LEDs
2.....Kabelverschraubung (M20) für Spannungsversorgung, Zustimmungstaster und Not-Aus
3.....Kabelverschraubung (M16) für separate Funktionserde (Auslieferungszustand mit Blindstopfen)
4.....Coninvers-Buchse für KeTop T100, T50 und T40 - Anschlusskabel
5.....Kabelerschraubung (M16) für Anschluss an Datenleitung
6.....Kabelverschraubung (M16) - Reserve (Auslieferungszustand mit Blindstopfen)

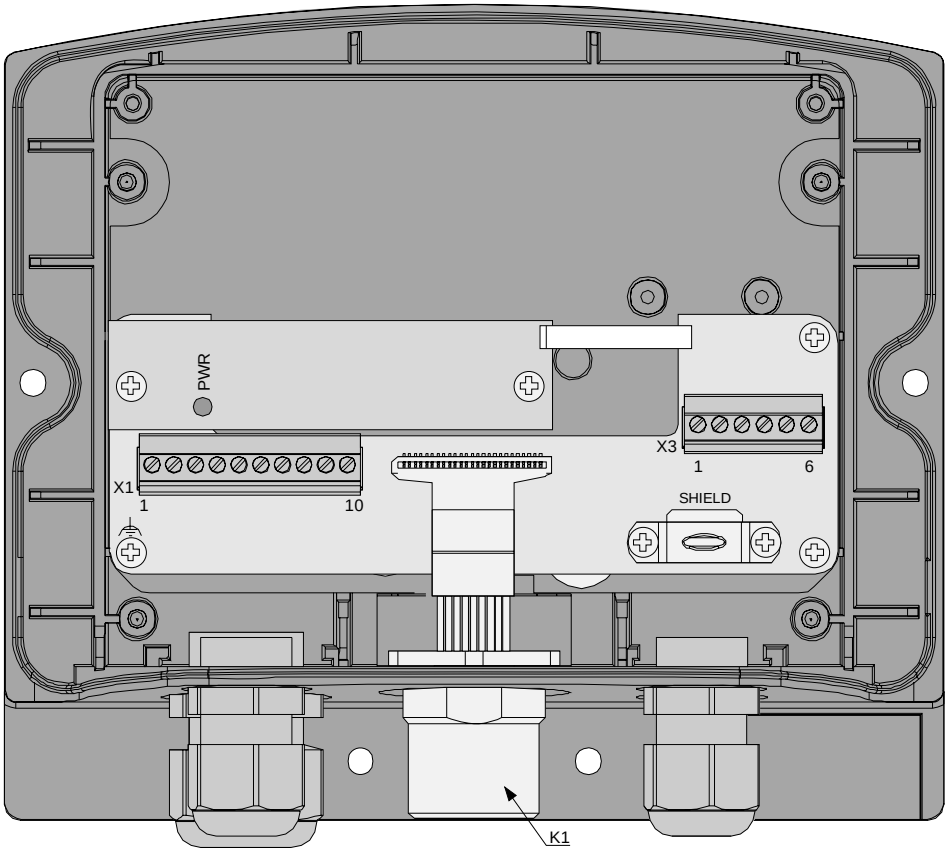
Ansichten und Gerätebeschreibung der Anschlussbox

⚠ WARNUNG

- **Die Anschlussbox und das Handterminal entsprechen der Schutzklasse III nach EN61131-2 und EN50178. Beim Anschluss ist daher folgendes zu beachten: Alle in die Anschlussbox oder in das Handterminal eingespeisten Spannungen müssen Sicherheitskleinspannungen sein und daher mit einem Sicherheitstrafo oder Gleichwertigem vom Niederspannungsnetz getrennt sein.**

© KEBA 2006

Innenansicht



K1 .....17-poliger Rundstecker (Coninvers) für KeTop-Anschlusskabel
X1 .....Klemmleiste für Versorgungs- und Steuerleitungen (Not-Aus, Zustimmungstaster)
X3 .....Klemmleiste für Datenleitungen
SHIELD.....Schirmschelle mit Auflagefläche für Daten-Kabelschirm
(Dient nicht zur Zugentlastung des Kabels!)

Innenansicht der Anschlussbox

Details bezüglich Anschließen der Anschlussbox KeTop CB 211 siehe jeweils Unterkapitel "Anschlussplan" der Kapitel "Ethernet" und "RS-422-A".

KEBA © KEBA 2006

Technische Daten der Anschlussklemmen

Für die in der Anschlussbox bereits vorhandenen PHOENIX-Stecker-Klemmleisten X1, und X3 gelten folgende technische Daten:

Anschlussvermögen:		
starr / flexibel / Leitergrößen	[mm²]/[mm²]/AWG	0,14-1,5 / 0,14-1,5 / 28-16
flexibel mit Aderendhülsen o. / m. Kunststoffhülse	[mm²]	0,25-1,5 / 0,25-0,5
Rastermaß:		3,81
Abisolierlänge:	[mm²]	7
Anzugsmoment:	[Nm]	0,22-0,25

PHOENIX-Bestelldaten:

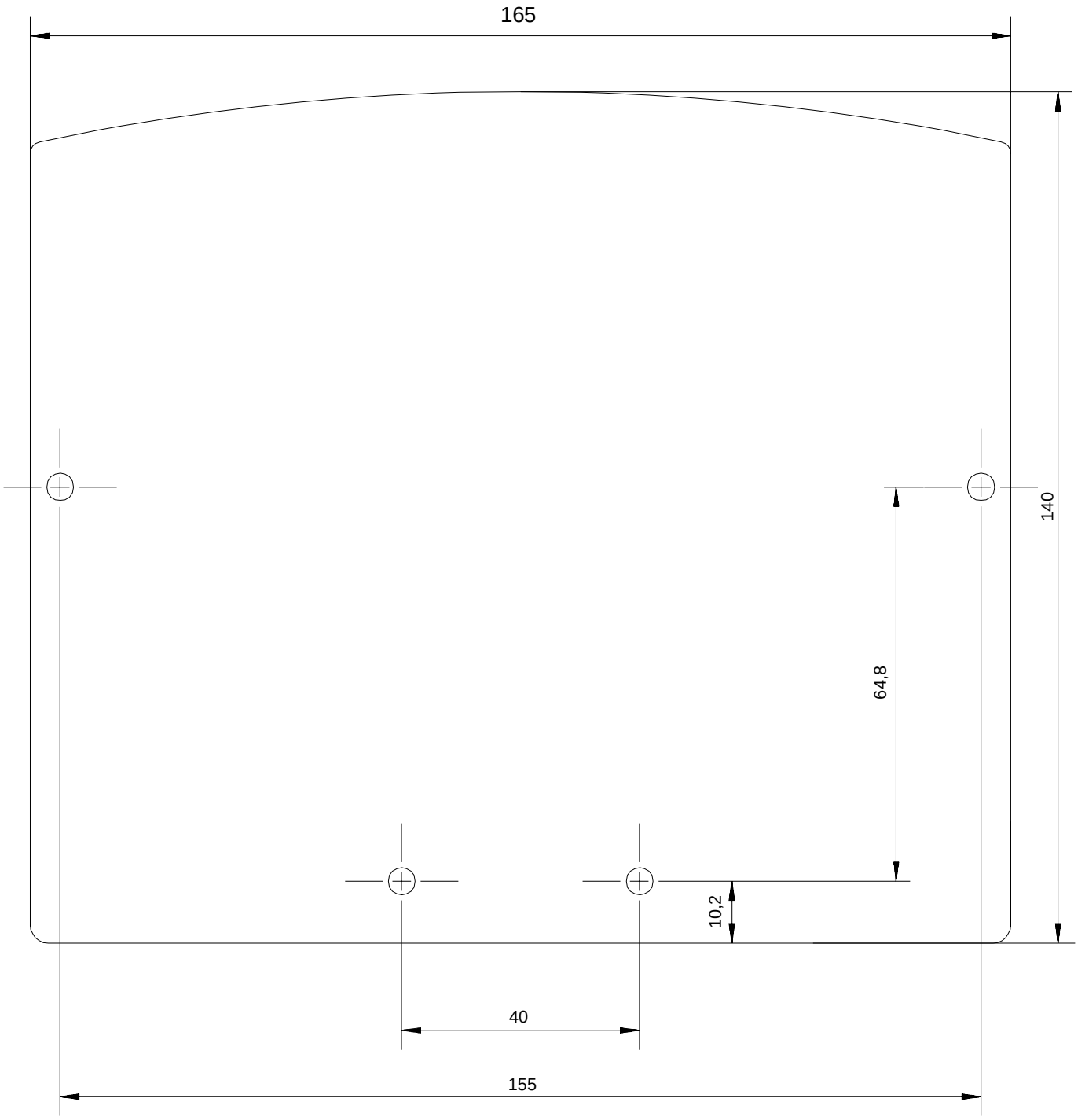
Gateway-Klemmleiste	PHOENIX Typ	PHOENIX Artikel-Nr.
X1	MCVR 1,5/6-ST-3,81	1827169
X3	MCVR 1,5/7-ST-3,81	1827172
X4, X4B	MCVR 1,5/10-ST-3,81	1827208

Hinweise

- *Beachten Sie bei der Wahl des Anschlusskabels das Anschluss-vermögen der Klemmleisten.*
- *Verwenden Sie für die Klemmleisten folgenden Schraubendreher zum Anklemmen der Adern:*
  Klinge: 0,4 x 2,5 x 80 mm,
  Länge: 160 mm
- *Es ist kein Mehrleiteranschluss (2 Leiter in einer Klemme) zulässig. Verwenden Sie zur Feldbusweiterführung die Klemmleiste X4B.*

© KEBA 2006

Bohrschablone für Wandmontage



Bohrschablone für die Gatewaybox, Maßstab 1:1

© KEBA 2006

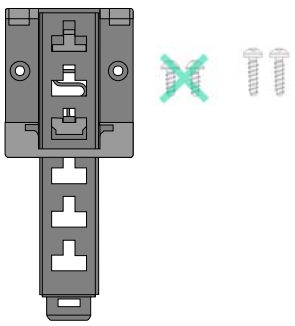
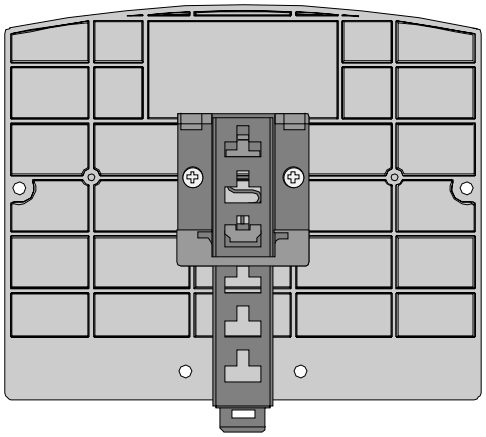
Für die Wandmontage empfehlen wir folgendes Schraubenmaterial:

- Spanplattenschraube Ø 4 x 40mm
  Kopfform: Flachkopf
  Max. Kopfdurchmesser: Ø 9mm

- Empfohlener Dübel: Ø 6 x 30mm

Hutschienen-Montagesatz KeTop DR200

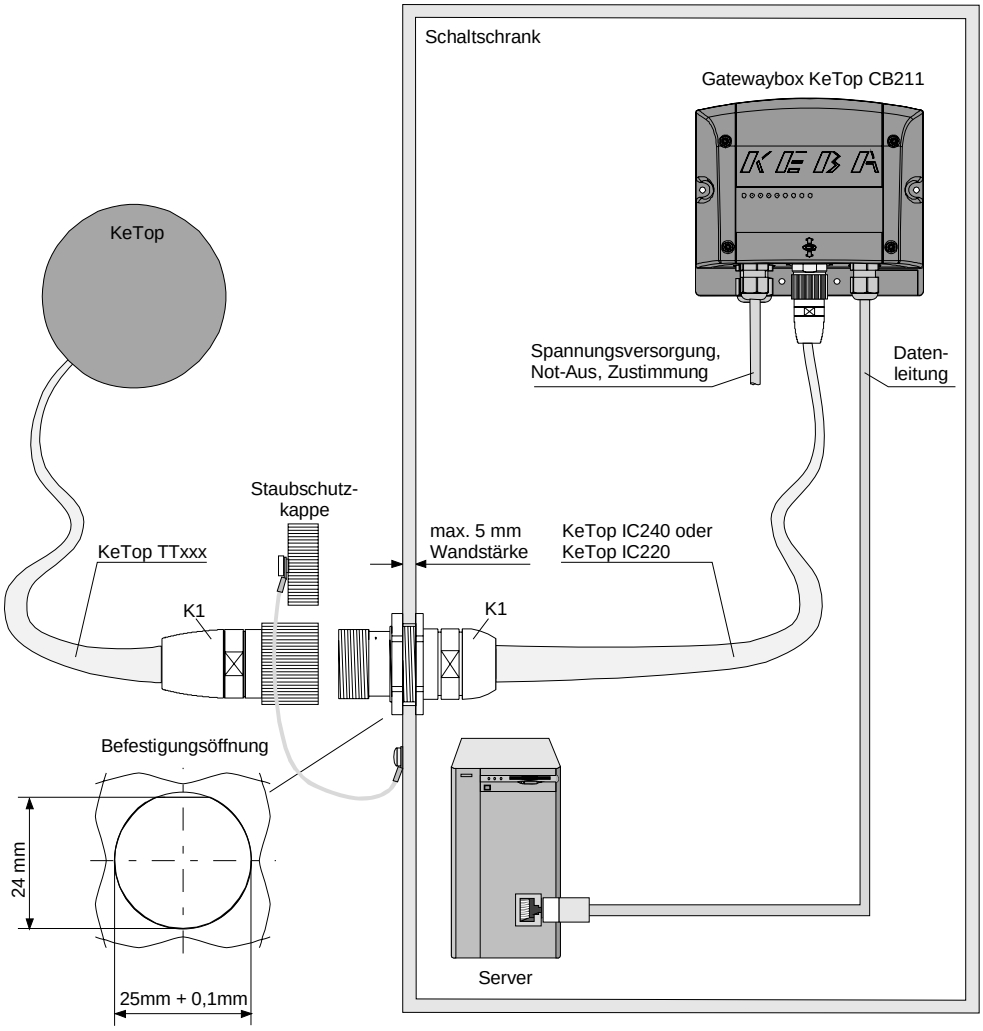
Der Hutschienenhalter KeTop DR200 ist als Zubehör erhältlich und wird auf der Gatewaybox KeTop CB2xx rückseitig montiert. Dadurch kann die Gatewaybox KeTop CB2xx einfach auf einer Hutschiene aufgeschnappt werden.

	
Hutschienen-Montagesatz	Hutschienenhalter an der Gatewaybox-Rückseite montiert

Hutschienen-Montagesatz für die Gatewaybox

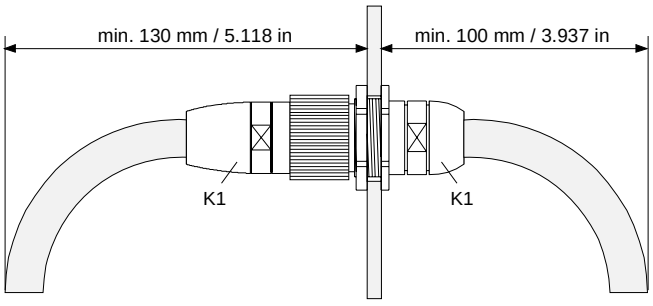
© KEBA 2006

Verwendung der Anschlussbox



Anschlussbox KeTop CB211 im Schaltschrank

Minimaler Kabelbiegeradius



Benötigter Minimalabstand außer- und innerhalb des Schaltschrankes

© KEBA 2006

Technische Daten Gatewaybox KeTop CB211

Allgemein

Versorgungsnennspannung:	24 V DC
Versorgungsspannungsbereich:	18 V DC bis 32 V DC
Max. Unterbrechungsdauer der	
Versorgungsspannung:	≤ 10 ms (lt. EN 61131)
Leistungsaufnahme:	
ohne KeTop:	3,6 W (200 mA bei 18 V DC, 150 mA bei 24 V DC)
mit KeTop:	10,8 W (600 mA bei 18 V DC,450 mA bei 24 VDC)
Einschaltstrom:	max. 5,6 A (Strombegrenzung vorhanden)
Schutzklasse:	III nach EN 61131-2 bzw. EN 50178

Umgebungsbedingungen

Betriebstemperatur:	0 °C bis 50 °C
Lagertemperatur:	-20 °C bis +70 °C
Rel. Luftfeuchtigkeit (nicht kondensierend):	5 % bis 95 %
Vibrationsfestigkeit:	(IEC 60068-2-6)
	5 ≤ f < 9Hz 7 mm
	9 ≤ f ≤ 150Hz 2g
Schockfestigkeit:	15 g / 11 ms (IEC 60068-2-27)

Gehäuse

Konstruktion	
	Gehäuse aus ABS
	Beständig gegen Fette, Öle, Schmierstoffe, Alkohol u.a.
Flammwidrigkeit:	UL94-V0
Abmessungen:	
Breite:	160 mm
Höhe:	140 mm
Tiefe:	70 mm
Schutzart:	IP65
Gewicht:	500g
Anzeige	Status LEDs

Zubehör

Zwischenkabel	Verbindung zwischen Anschlussbox und Anschlusskabel
KeTop IC220:	2 m
KeTop IC240:	4 m
Downloadkabel	
KeTop XD040:	4 m, zum Download von SW und zur Fehlersuche via S2 im KeTop
Hutschienen-Montagesatz	
KeTop DR100	Hutschienenadapter für CB2xx
Brückenstecker	
KeTop BC001:	Dient bei abgestecktem KeTop zur Überbrückung der Not-Aus-Kreise

© KEBA 2006

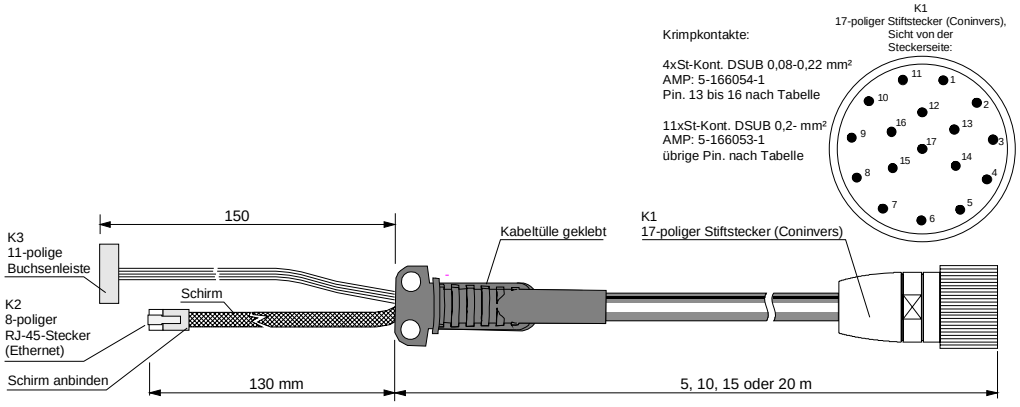
Anschlusskabel KeTop TTxxx

Die KeTop-Handterminals sind standardmäßig mit folgenden Anschlusskabeln erhältlich:

- KeTop TT025 (2,5 m)
- KeTop TT050 (5 m)
- KeTop TT100 (10 m)
- KeTop TT150 (15 m)
- KeTop TT200 (20 m)

Das Anschlusskabel ist beständig gegen Wasser, Reinigungsmittel (Alkohole und Tenside), Öle, Schneideöle (Bohröle), Fette und Schmierstoffe.

Signal-Beschreibung	K3, 11-polige Buchsen-leiste zu S19 am KeTop	K2, 8-poliger RJ-45-Stecker (S4, Ethernet)	Anschlusskabel KeTop TTxxx, Adernfarbe		K1, 17-poliger Stift-stecker, Pin-Nr.:
24 VDC	6	-	rosa	->	1
GND_IN	7	-	schwarz	->	2
Not-Aus, Kreis 1	8	-	braun-grün	->	3
Not-Aus, Kreis 1	9	-	weiß-grün	->	4
Not-Aus, Kreis 2	10	-	grau-rosa	->	5
Not-Aus, Kreis 2	11	-	rot-blau	->	6
Zustimmung, Kreis 1, pos.	1	-	braun	->	7
Zustimmung, Kreis 1, neg.	2	-	gelb	->	8
Zustimmung, Kreis 2, pos.	3	-	grün	->	12
Zustimmung, Kreis 2, neg.	4	-	grau	->	17
nicht verwendet	n.c.	-	-	-	9
nicht verwendet	n.c.	-	-	-	10
nicht verwendet	5	-	violett	->	11
TD+ I CAN+	-	1	blau	->	13
TD- I CAN-	-	2	weiß	->	14
RD+ I SGND	-	3	orange	->	15
RD- I nicht verw.	-	6	rot	->	16



Anschlusskabel KeTop TTxxx

KEBA © KEBA 2006

Zwischenkabel KeTop IC2xx

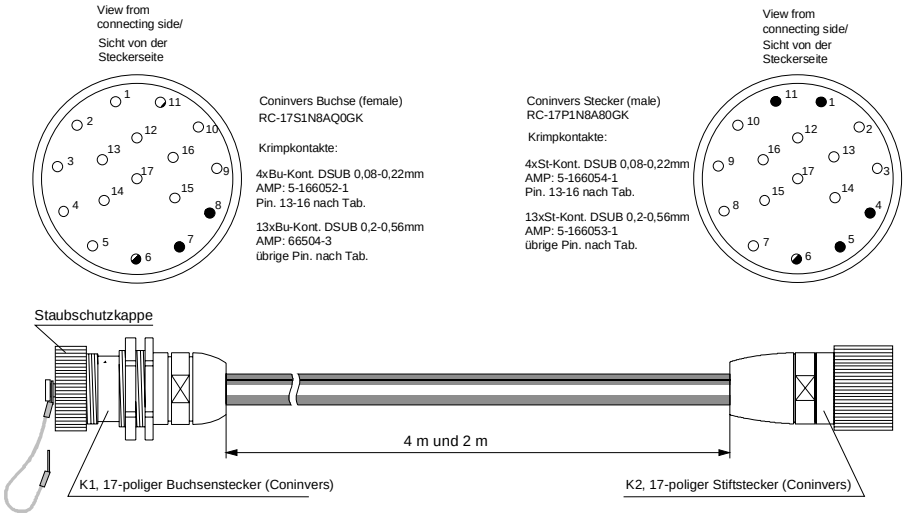
Für die KeTop-Handterminals sind folgenden Zwischenkabel erhältlich:

- KeTop IC240 (4 m)
- KeTop IC220 (2 m)

Das Zwischenkabel ist beständig gegen Wasser, Reinigungsmittel (Alkohole und Tenside), Öle, Schneideöle (Bohröle), Fette und Schmierstoffe.

Signal-Beschreibung	K1, 17-poliger Buchsen-stecker, Pin-Nr.:	Zwischenkabel KeTop IC2xx, Adernfarbe		K2, 17-poliger Stiftstecker, Pin-Nr.:
Zustimmung, Kreis 1, pos.	7	braun	<-	7
Zustimmung, Kreis 1, neg.	8	gelb	<-	8
Zustimmung, Kreis 2, pos.	12	grün	<-	12
Zustimmung, Kreis 2, neg.	17	grau	<-	17
n.c.	10	n.c.		10
24 V DC (+/- 10%)	1	rosa	<-	1
n.c.	9	n.c.		9
Not-Aus Kreis 1	3	braun/grün	<-	3
Not-Aus Kreis 1 GND	4	weiß/grün	<-	4
Not-Aus Kreis 2	5	grau/rosa	<-	5
Not-Aus Kreis 1 GND	6	rot/blau	<-	6
GND_IN	2	schwarz	<-	⏤ 2
TD+ (Transmitt)	13	blau	<-	13
TD- (Transmitt)	14	weiß	<-	14
RD+ (Receive)	15	orange	<-	15
RD- (Receive)	16	rot	<-	16
Powerfail	11	violett	<-	11

Signale in der Tabellenumrahmung sind geschirmt



Zwischenkabel KeTop IC2xx

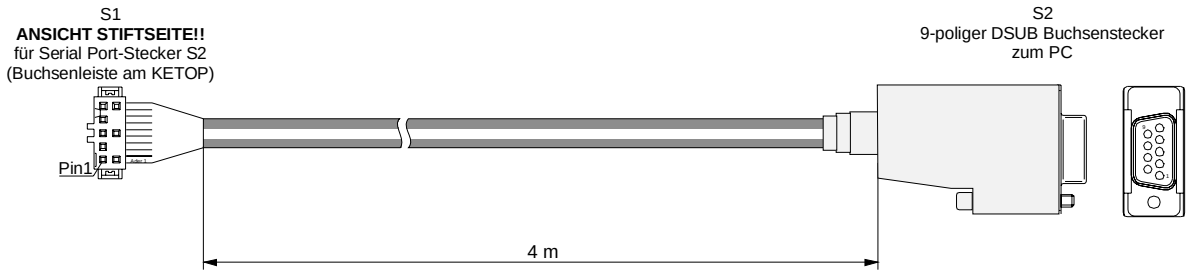
© KEBA 2006

Downloadkabel KeTop XD040

Das Downloadkabel KeTop XD040 ist mit einer Länge von 4 m erhältlich und wird an den Serial Port-Stecker S2 im Anschlussschacht des KeTops oder am X6 der Gatewaybox KeTop CB23x angeschlossen.
Es dient zum Downloaden von Software und zur Fehlersuche.

Handterminal KeTop xxxx		Gatewaybox KeTop CB23x		Downloadkabel KeTop XD040			PC
Signale	S2 Serial-Port-Stecker	Signale	X6 Serial-Port-Stekker	S1 Stiftleiste	Signal-Beschreibung	S2, 9-poliger DSUB Buchsenstecker	Signale (COMx)
-	1*	n.c.	1	1	< - >	1	(ActiveSync)
-	2*	n.c.	2	2	< - >	6	(ActiveSync)
TXD	3	TXD	3	3	< - >	2	RXD
CTS	4	n.c.	4	4	< - >	-	-
RXD	5	RXD	5	5	< - >	3	TXD
RTS	6	n.c.	6	6**	< - >	-	-
-	7*	n.c.	7	7	< - >	4	(ActiveSync)
n.c.	8	n.c.	8	8**	< - >	9	-
GND	9	GND	9	9	< - >	5	GND
GND	10	n.c.	10	10	< - >	-	nicht verw.
						7	nicht verw.
						8	nicht verw.

*) .....Die Pins 1, 2 und 7 des Serial Port-Steckers S2 sind direkt auf der Platine im KeTop gebügelt. Dh. es darf bei Selbstanfertigung eines Downloadkabels nicht auf diese 3 Adern im Kabel verzichtet werden, da sie für das ActiveSync-Signal benötigt werden.
**) .....Stift wurde von Stiftleiste entfernt (Versteckschutz)



Serielles Download-Kabel XD040

KEBA © KEBA 2006

11Transportbedingungen

Damit das Gerät bei einem Weiter- oder Rücktransport keinen Schaden nimmt, müssen folgende Transportbedingungen eingehalten werden:

- Verwenden Sie für den Transport immer die Originalverpackung.
- Die Umgebungsbedingungen für das Gerät (siehe Kap. „Technische Daten“) müssen auch während des Transportes eingehalten werden.

© KEBA 2006

12Technische Daten

Allgemein

Versorgungsnennspannung:	24 V DC (Sicherheitskleinspannung)
Versorgungsspannungsbereich:	18 V DC bis 32 V DC
Max. Unterbrechungsdauer der Versorgungsspannung:	≤ 10 ms (lt. EN 61131)
Leistungsaufnahme:	7,2 W (400 mA bei 18 V DC, 300 mA bei 24 V DC)
Einschaltstrom:	max. 5,6 A (Strombegrenzung vorhanden)
Schutzklasse:	III nach EN 61131-2 bzw. EN 50178

Umgebungsbedingungen

Betriebstemperatur:	0 °C bis 50 °C
Lagertemperatur:	-20 °C bis +70 °C
Relative Luftfeuchtigkeit (nicht kondensierend):	5 % bis 95 %
Vibrationsfestigkeit:	(IEC 60068-2-6)
	10 Hz ≤ f < 57 Hz mit 0,15 mm
	57 Hz ≤ f < 150 Hz mit 2 g
Schockfestigkeit:	25 g / 11 ms (IEC 60068-2-27)

Gehäuse

Konstruktion	
	Doppelwandiges Gehäuse aus ABS
	Beständig gegen Fette, Öle, Schmierstoffe, Alkohol u.a.
Flammwidrigkeit:	UL94-V0
Abmessungen:	
Durchmesser:	290 mm
Gesamthöhe:	130 mm
Schutzart:	IP54
Gewicht:	1800 g (mit Not-Aus, ohne Handrad, ohne Override Poti, ohne Kabel)
	+ 100 g für PCMCIA-Erweiterung
	+ 20 g für RS-422-A-Zusatzmodul
Farbdisplay 7,7“	
Typ:	grafikfähiges STN-LC-Display
Größe:	7,7“ (160 x 120 mm)
Auflösung:	VGA 640 x 480 Pixel
Darstellung:	256 Farben
Hintergrundbeleuchtung:	CCFT-Kaltkathodenröhre (40.000h Betriebsdauer)
Touch-Screen:	Analog-resistiv
Farbdisplay 8,4“	
Typ:	grafikfähiges TFT-Display
Größe:	8,4“ (175 x 130 mm)
Auflösung:	SVGA 800 x 600 Pixel
Darstellung:	65535 Farben
Hintergrundbeleuchtung:	CCFT-Kaltkathodenröhre (40.000h Betriebsdauer)
Touch-Screen:	Analog-resistiv
Tastatur:	
	- Folientasten mit taktiler Rückmeldung
	- Tastenfelder für Daumenbedienung ausgelegt
	- Bedienung für Rechts- und Linkshänder
	- Kundenspezifische Tastatur möglich
	- 2 Status LEDs
Bedienelemente:	
	zwei 3-stufige Zustimmungstaster, 2-kreisig

KEBA © KEBA 2006

Optionen:	
	Override Potentiometer
	Elektronisches Handrad
	Joystick
	Not-Aus-Taster
2 Einbauplätze für:	Leuchtdrucktaster, tastend 0
	Leuchtdrucktaster, tastend I
	Leuchtdrucktaster, rastend I/0
1 Einbauplatz für:	Leuchtdrucktaster, tastend 0
	Leuchtdrucktaster, tastend I
	Leuchtdrucktaster, rastend I/0
	Schlüsselschalter, 3 Stellungen: I-0-II / 0-I-II
	Wahlschalter, 3 Stellungen: I-0-II / 0-I-II

Rechner und Schnittstellen

Prozessor:	Intel StrongARM SA-1110/206 MHz
Speicher:	DRAM: max. 128 MB, FLASH: 64 MB
Schnittstellen:	Ethernet
	RS-422-A
	RS-232-C (Debug-Schnittstelle im Gerät)
	PC-Card-Slot (PC-Card Typ I, II, III)
Betriebssytem:	Windows CE 4.2

Zubehör

Wandhalterung	zum stationären Betrieb oder zur Ablage
KeTop WB090:	Wandhalterung ohne Kabelhalterung
KeTop WB095:	Wandhalterung mit Kabelhalterung
KeTop WB100:	Wandhalterung mit Höhenverstellung und ohne Kabelhalterung
KeTop WB110:	Wandhalterung mit Höhenverstellung und mit Kabelhalterung
Anschlusskabel	Torsions-, biege- und trittfest, mit konfektionierten 20 pol. Push Pul Stecker oder 17-pol. Rundstecker
KeTop TT025:	2,5 m
KeTop TT050:	5 m
KeTop TT100:	10 m
KeTop TT150:	15 m
KeTop TT200:	20 m
Zwischenkabel	Verbindung zwischen Anschlussbox und Anschlusskabel
KeTop IC220:	2 m
KeTop IC240:	4 m
Downloadkabel	(nicht für KeTop T30)
KeTop XD040:	4 m, via S2 im KeTop bzw. via X6 an der KeTop CB23x
Anschlussboxen	(KeTop CB23x nicht für KeTop T30)
KeTop CB211:	Zur Auftrennung der Daten- und Steuerleitungen
KeTop CB230:	Gatewaybox zum Anschluss an CAN über die RS-422-A-Schnittstelle
KeTop CB232:	Gatewaybox zum Anschluss an CANopen über die RS-422-A-Schnittstelle
KeTop CB233:	Handterminal an DeviceNET über die RS-422-A-Schnittstelle
KeTop CB234:	Gatewaybox zum Anschluss an INTERBUS über RS-422-A-Schnittstelle
KeTop CB235:	Gatewaybox zum Anschluss an PROFIBUS-DP über RS-422-A-Schnittstelle
KeTop CB236:	Gatewaybox zum Anschluss an PROFIBUS-MPI über RS-422-A-Schnittstelle
Projektierungssoftware	Nur für KeTop T100 und KeTop T50
KeTop PT001:	zenOn 5.50
Start-Kit	
KeTop SK001:	Handbücher englisch/deutsch auf CD
	SDK für Windows CE

© KEBA 2006

	Projektierungstool KeTop PS040
	Demoapplikationen zenOn 5.50
	Demoapplikation Steuerungen

Ersatzteile

Touchstifte	
KeTop E-TP001:	3 Stück original KeTop Touchstifte
Anschlussschachtdeckel	
KeTop E-SD100:	mit Dichtung und Befestigungsmaterial für das KeTop T100.
PC-Card-Deckel	
KeTop E-PC100:	mit Dichtung und Befestigungsmaterial für das KeTop T100.

© KEBA 2006

13CE Konformität, Richtlinien und Normen

Richtlinien der Europäischen Union

Ein wesentliches Ziel der Europäischen Union ist die Realisierung eines europäischen Binnenmarktes, und damit verbunden der Abbau von Handelshemmnissen.
Um dieses Ziel zu erreichen, werden durch die Europäischen Verträge die „vier Freiheiten“ gewährleistet:

- Freier Warenverkehr
- Niederlassungsfreiheit
- Freier Dienstleistungsverkehr
- Freier Kapitalverkehr

Der freie Warenverkehr bedeutet, dass quantitative Einfuhrbeschränkungen von Waren zwischen den Mitgliedstaaten verboten sind.

Ausgenommen davon sind Waren, die die Sicherheit von Personen oder der Umwelt gefährden. Solche Produkte können von Mitgliedsstaaten auf ihrem Hoheitsgebiet unterbunden werden.
Um auch für diese Produkte den freien Warenverkehr sicherzustellen werden die nationalen Sicherheitsbestimmungen der Mitgliedsstaaten mittels Richtlinien der Europäischen Union vereinheitlicht.
Diese Richtlinien gibt es für eine Reihe von Produktklassen, wie z.B.: Maschinen, Medizinprodukte oder auch Spielwaren. Aber auch für weitere gemeinsame Sicherheitsaspekte von Produkten, wie Schutz vor Elektrizität, Explosionsschutz oder Elektromagnetische Verträglichkeit wurden entsprechende Richtlinien erarbeitet.

Die Richtlinien richten sich an die Mitgliedsstaaten, welche diese in den jeweiligen nationalen Gesetzen umsetzen müssen. Die Richtlinien haben daher Gesetzescharakter.

Mit der „CE“-Kennzeichnung bescheinigt der Hersteller alle Verpflichtungen der auf das Produkt zutreffenden EU-Richtlinien erfüllt zu haben.
Das CE-Zeichen, welches der Hersteller selbst auf die Produkte aufbringt, ist der „Reisepass“ innerhalb der EU und für die überwachenden Behörden bestimmt.
Ergänzend dazu kann von uabhängigen, akkreditierten Zertifizierungsstellen die Konformität mit den EU-Richtlinien überprüft und dies mit einer EG-Baumusterbescheinigung bestätigt werden.

Für Handterminals ist neben der EMV Richtlinie (EMV RL 89/336/EWG) auch die Maschinen Richtlinie (MRL 98/37/EG) anzuwenden.

© KEBA 2006

Maschinen Sicherheit

Überall, wo an der Maschine auftretende Fehler Personenschaden oder große Materialschaden verursachen können, müssen zusätzliche Maßnahmen getroffen werden, die auch im Fehlerfall einen sicheren Betriebszustand des Gesamtsystems gewährleisten.
Obwohl das Handterminal selbst keine Maschine im engeren Sinn ist, erfüllt es doch wesentliche Aufgaben zur Gewährleistung von Sicherheitsfunktionen einer Maschine oder Anlage an die es angeschlossen ist.
Das Handterminal verfügt z.B.: über die Sicherheitsfunktionen „Stoppen im Notfall“ und über eine Zustimmungseinrichtung für Bedienung in Sonderbetriebsarten, und ist somit ein „Sicherheitsbauteil“ im Sinne der Maschinenrichtlinie (MRL).
Sicherheitsbauteile, also Teile deren Ausfall oder Fehlfunktion die Sicherheit von Personen im Gefahrenbereich der Maschine gefährden, fallen ausdrücklich in den Anwendungsbereich der MRL.
Wesentliche Anforderungen der MRL an den Hersteller einer Maschine oder Anlage sind:

- Durchführung einer Gefahren- und Risikoanalyse
- Befolgung der Grundsätze für die Integration der Sicherheit
- Erstellung und Verwahrung einer Technischen Dokumentation
- Lösungen gemäß dem Stand der Technik
- Konformitätsvermutung mittels Harmonisierten Normen
- Aufbringung der CE-Kennzeichnung

Für Sicherheitsbauteile gelten im wesentlichen die gleichen Anforderungen. Für Sicherheitsbauteile muss nachgewiesen werden, dass ein Ausfall oder eine Fehlfunktion nicht möglich ist, oder, dass eine Fehlfunktion nicht zu einer gefährlichen Situation führt.

„Gefahren- und Risikoanalyse“

Der Hersteller einer Maschine ist verpflichtet, seine Maschine in allen Lebensphasen und Betriebsarten zu analysieren, und alle möglichen auftretenden Gefahren zu dokumentieren. Dies geschieht ohne Berücksichtigung von eventuellen Schutzeinrichtungen.
Als nächster Schritt wird für jede erkannte Gefahr ein Schutzziel formuliert, und anschließend eine oder mehrere Schutzmaßnahme(n) zu Erreichung des Schutzzieles definiert.

Weitere Details zur Vorgangsweise bei der Durchführung der Gefahren- und Risikoanalyse sowie Listen von häufig anzutreffenden Gefährdungen finden sich in folgenden Normen:

- EN 292-1 und EN 292-2 „Allgemeine Gestaltungsleitsätze von Maschinen“
- EN 1050 „Leitsätze zur Risikobeurteilung von Maschinen“

KEBA © KEBA 2006

„Grundsätze für die Integration der Sicherheit“

Im Anhang I, Kap. 1.1.2 der MRL 98/37/EG schreibt die MRL eine klare Vorgangsweise und Reihenfolge für die Auswahl der Schutzmaßnahmen vor:

Beseitigen oder Minimieren der Gefahren

Dies geschieht bereits durch die Konstruktion der Maschine
Unter diese Maßnahmen fallen z.B.:

- Reduktion von auftretenden Energien (Kräfte, Drehzahlen, Spannungen…), soweit dies möglich ist
- Vermeidung unnötiger Scherstellen oder scharfen Kanten
- Vermeidung von Fehlbedienungen durch ergonomische und logische Gestaltung von Bedienvorrichtungen
- Vermeidung gefährlicher Materialien und Verbrauchsstoffe

Treffen von Schutzmaßnahmen gegen Gefahren, die sich nicht beseitigen lassen

Unter diese Maßnahmen fallen z.B.:

- Trennende Schutzeinrichtungen (Umwehrungen, Gehäuse, Schutzgitter)
- Nicht trennende Schutzeinrichtungen (Lichtschranken zur Abschaltung der Gefahren)
- Steuerungstechnische Schutzeinrichtungen (Zustimmeinrichtungen, Zweihandschaltungen, Drehzahlüberwachungen…)

Unterrichtung des Benutzers über Restgefahren

Diese letzte der drei Möglichkeiten wird angewendet, wenn trotz der ersten beiden Punkte Restgefahren bestehen bleiben. Unter diese Maßnahmen fallen z.B.:

- Warnhinweise
- Schulungen und organisatorische Maßnahmen
- Verwendung persönlicher Schutzausrüstung

„Technische Dokumentation“

Die Technische Dokumentation beinhaltet alle Dokumente, welche zum Nachweis der Sicherheit der Maschine / des Sicherheitsbauteils erforderlich sind. Das sind z.B.:

- Gesamtplan der Maschine oder des Sicherheitsbauteils inklusive Steuerkreispläne
- Gefahren- und Risikoanalyse
- Berechnungen
- Versuchs- und Testergebnisse

© KEBA 2006

- Liste der grundlegenden für die Maschine anzuwendenden Sicherheitsanforderungen der MRL und Beschreibung der Lösungen
- Angewandte Normen
- Bedienungs- und Betriebsanleitungen
- Auflistung der QS-Maßnahmen in der Serie

Die Technische Dokumentation muss noch bis mindestens 10 Jahre nach Auslieferung des letzten Produktes aufbewahrt werden, und in einem Schadensfall innerhalb einer Frist von wenigen Tagen vorgelegt werden.

„Stand der Technik“

Darunter werden technische Möglichkeiten zu einem bestimmten Zeitpunkt verstanden, welche auf gesicherten Erkenntnissen von Wissenschaft und Technik basieren. Der Stand der Technik beinhaltet auch, dass er wirtschaftlich durchführbar ist, d.h., von der Mehrheit in dem betreffenden industriellen Sektor geleistet werden kann.
Stand der Technik ist der Entwicklungsstand fortschrittlicher Verfahren, Einrichtungen oder Betriebsweisen, der die praktische Eignung der Maßnahme im Hinblick auf die angestrebten Ziele (z.B. der Ziele des Arbeitsschutzes, des Umweltschutzes, der Sicherheit für Dritte, der Wirtschaftlichkeit: Also allgemein zur Erreichung eines allgemein hohen Niveaus bezogen auf die zu beachtenden Aspekte) insgesamt gesichert erscheinen lässt.
Der Stand der Technik kann sich unabhängig von den Normen weiterentwickeln.

„Konformitätsvermutung mit Harmonisierte Normen“

Die Europäischen Richtlinien beinhalten hauptsächlich allgemeine Anforderungen an die Sicherheit von Produkten, jedoch keine Realisierungsdetails. Hierfür sind die europäischen Normungsinstitute zuständig, welche für konkrete sicherheitstechnische Problemstellungen oder bestimmte Produktklassen Vorschläge für die Umsetzung liefern. Normen, bei denen davon ausgegangen wird, dass sie die Anforderungen der Richtlinien treffen und sinngemäß richtig umsetzen, werden als „Harmonisierte Normen“ aufgelistet. Bei weitem nicht alle erhältlichen Normen sind jedoch harmonisiert.

Ein Hersteller kann durch Anwendung und Umsetzung von Harmonisierten Normen die Konformitätsvermutung des jeweiligen Produktes aussprechen. Trotzdem sind die Normen, im Gegensatz zu den Richtlinien, nicht rechtlich verpflichtend. Das bedeutet, dass der Hersteller auch andere Lösungen, als die in den Normen beschriebenen, in Betracht ziehen darf. Allerdings müssen diese Lösungen mindestens das gleiche Sicherheitsniveau wie die relevanten Normen erreichen und den Anforderungen der zutreffenden Richtlinien genügen.

KEBA © KEBA 2006

Auswahl von Sicherheits-Kategorien nach EN 954-1

Die Maschinenrichtlinie fordert, dass ein Defekt in der Logik des Steuerkreises, oder aber auch eine Störung oder Beschädigung desselben, nicht zu einer gefährlichen Situation führen darf.
Dieser allgemeine Ansatz wird in der EN 954-1 „Sicherheitsbezogene Teile von Maschinensteuerungen“ konkretisiert, welche für sicherheitsrelevante Steuerungsteile Sicherheits-Kategorien (B,1,2,3,4) definiert.
Diese Kategorien gelten dabei unabhängig von der eingesetzten Technologie also z.B. für elektrische, elektromechanische oder auch für hydraulische oder pneumatische Systeme.

Die Kategorien stellen qualitative Anforderungen an die Ausfallswahrscheinlichkeit, die Erkennung von Fehlern und das Verhalten der Steuerung im Fehlerfall.
Die Auswahl der Kategorie erfolgt vom Hersteller der Maschine abhängig vom tatsächlichen Gefahrenpotential, welche anhand der Gefahren- und Risikoanalyse ermittelt wird.
Bei Gefahren, welche irreversible Verletzungen oder Tod zur Folge haben können, ist üblicherweise mindestens Sicherheits-Kategorie 3 oder 4 erforderlich. Diese Kategorien erfordern bereits eine 1-Fehler Sicherheit, was in klassischer Weise meistens mit mehrkreisiger Technik umgesetzt wird.
Wichtig ist in diesem Zusammenhang auch, dass einzelne Fehler rechtzeitig erkannt werden müssen, um eine Fehleranhäufung, welche schließlich zum Sicherheitsverlust führen kann, zu vermeiden. Fehler, welche erkannt werden müssen, sind bei elektrischen und elektronischen Systemen z.B.: Querschlüsse zwischen den Kreisen, Unterbrechungen, Kurzschlüsse oder verklebte Kontakte.

Häufig werden zur Erkennung von Fehlern in den einzelnen Sicherheits-Kreisen spezielle zertifizierte Sicherheits-Schaltgeräte eingesetzt. Die bei diesen Geräten angegebene Sicherheits-Kategorie wird jedoch nur dann erreicht, wenn auch die Beschaltung mit den zugehörigen Kreisen der jeweiligen Kategorie entspricht. Die Sicherheits-Kategorien müssen immer für eine gesamte Sicherheitsfunktion betrachtet werden, nicht für einzelne Komponenten oder Bauteile.

Der Nachweis für die Erreichung einer Sicherheits-Kategorie kann mit Hilfe einer FMEA (Failure Mode and Effects Analysis) erfolgen, bei der alle möglichen anzunehmenden Fehler theoretisch oder auch praktisch durchgespielt werden, und gezeigt wird, dass den Anforderungen der Kategorie genüge getan ist.

Anwendung von Handterminals in Sonderbetriebsarten

Bei der manuellen Steuerung von Maschinen in Sonderbetriebsarten, wo die Sicherheit vom rechtzeitigen Reagieren des Bedienpersonals abhängt, ist es unbedingt erforderlich, dass der Bedienbereich vom Bediener eingesehen werden kann.
Das Handterminal bietet hier den Vorteil, dass damit sehr nahe an den Bedienbereich herangetreten werden kann.

© KEBA 2006

Gleichzeitig steigt mit der Mobilität die Missbrauchgefahr dadurch, dass mit dem Handterminal auch an entfernteren Orten, wo der Bedienbereich nicht mehr wahrgenommen werden kann, bewusst oder unbewusst Maschinenbewegungen in Gang gesetzt werden können.
Der Betreiber der Maschine hat daher durch Auswahl der entsprechenden Kabellänge des Handterminals für den richtigen Kompromiss aus notwendiger Flexibilität und einer zumutbaren Arbeitsbereichsbeschränkung zu finden.
Bei Funk-Handterminals ist eine Arbeitsbereichseinschränkung durch das Kabel nicht möglich, daher sind bei diesen Handterminals zusätzliche technische Lösungen notwendig.

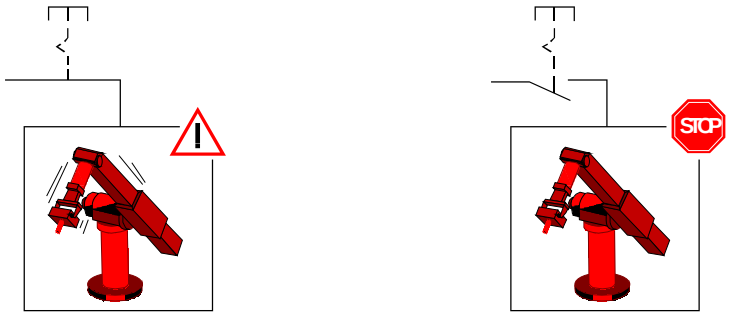
Wird die Maschine oder Anlage mit dem Handterminal betrieben, ist darauf zu achten, dass die Bedienung zu diesem Zeitpunkt ausschließlich durch das Handterminal und von keinem anderen Punkt der Anlage aus möglich ist.

Der Gefahrenbereich darf nur von derjenigen Person betreten werden, welche das Handterminal bedient, betreten werden. Sollte es notwendig sein, dass mehr als eine Person gleichzeitig im Gefahrenbereich arbeiten, benötigt jede anwesende Person eine Zustimmungseinrichtung, und die Bewegung darf nur freigegeben werden, wenn alle Zustimmungseinrichtungen betätigt werden.

Hinweise zum Schalter für das Stillsetzen im Notfall („Not-Aus“)

Theoretisch sollte eine perfekt konstruierte Maschine keinen Not-Aus benötigen, da die Maschinenrichtlinie fordert, dass eine Maschine in allen Lebensphasen und Betriebsarten sicher ist.
Aus der Praxis weiß man jedoch, dass es trotz aller Vorsichtsmaßnahmen immer wieder zu unvorhersehbaren Situationen kommt. Um in diesen Fällen ein rasches Stillsetzen der Maschine zu ermöglichen, bzw. die Gefahr zu bannen, wird an den meisten Maschinen ein Not-Aus vorgesehen.

Wie aus der nachfolgenden Abbildung ersichtlich ist, befindet sich eine Maschine aus Sicht des Not-Aus ständig im gefährlichen Zustand, solange dieser nicht betätigt ist.



Ruhestellung (nicht betätigt) "Not-Aus" gedrückt

Not-Aus-Funktionen

KEBA © KEBA 2006

Der Not-Aus darf daher nicht als vorwiegende Sicherheits-Einrichtung dienen, sondern ist ausschließlich zur Abdeckung von eventuellen Restrisiken vorgesehen.
Als primäre Sicherheits-Einrichtung sollten stattdessen, abhängig von der Betriebsart, z.B. verriegelnde Schutztüren, Lichtschranken, Zweihandschalter oder Zustimmungstaster angewendet werden.

Mit der Betätigung des Not-Aus Schalters muss die gesamte Maschine oder alle zu einer Anlage zusammen geschalteten Maschinen in einen sicheren Zustand übergeführt werden, zB durch Abschaltung der Energie der gefährdenden Antriebe (Stopp-Kategorien 0 oder 1 nach EN 60204-1).
Das Entriegeln des Not-Aus-Schalters darf keinen unkontrollierten Wiederanlauf der Maschine bewirken.

Unabhängig davon, ob auf einem Handterminal ein Not-Aus vorhanden ist oder nicht, sind in jedem Fall an ausgewählten Stellen rund um die Maschine fix installierte, für jedermann erkenntliche und gut zugängliche Not-Aus Schalter anzubringen.

Anwendung von Not-Aus Schalter bzw. Stopp-Schalter am Handterminal

Die Maschinenrichtline stellt keine detaillierten Anforderungen zur Farbgebung der Bedienelemente zum Stillsetzen im Notfall. Die Anforderung lautet: “Die Befehlseinrichtung muss deutlich kenntliche, gut sichtbare und schnell zugängliche Stellteile haben“.
Im Normalfall sind „Not-Aus“ Schalter rot-gelb gekennzeichnet. Die besondere Gestaltung hat den Sinn einer Signalwirkung, und soll bewirken, dass jeder, speziell auch ungeschulte Personen, in einem Notfall, das Bedienelement rasch als die Einrichtung zur Beseitigung einer Gefahr erkennen kann.

Eine unbedingte Forderung ist daher, dass mit diesen derart gekennzeichneten Geräten, zu jeder Zeit und in jeder Betriebsart, ohne weitere Kenntnis über die Maschine, ein sicherer Zustand eingeleitet werden kann (siehe dazu auch EN 418).
Eine wechselnde Betriebsbereitschaft eines Not-Aus-Schalters ist nicht zulässig, da dies in Paniksituationen zu Fehlhandlungen und lebensbedrohlichen Zeitverlusten führen kann.

Betrachtet man Handterminals in Bezug auf diese Forderungen, wird offensichtlich, dass abhängig von bestimmten Kriterien und Anwendungsfällen unterschieden werden muss, ob am Handterminal ein Not-Aus Schalter verwendet werden darf oder nicht.

Fix installierte Handterminals: Not-Aus Schalter

Diese sind mit einem Kabel ausgestattet, mit dem das Handterminal mit der definierten Maschine verbunden wird. Das Handterminal wird üblicherweise im ausgeschalteten Zustand der Maschine an die Maschine angeschlossen bzw. abgesteckt. Dies geschieht während des Installations- bzw. Deinstallationsvorganges. Die Handterminals sind nicht dazu vorgesehen, während des Betriebs der Maschine an- bzw. abgesteckt zu werden.

© KEBA 2006

Bei vielen einfachen Maschinen ist das Handterminal auch die einzige Bedienmöglichkeit, sodass die Maschine ohne Handterminal gar nicht in Betrieb gesetzt werden kann. Wird der Stecker dennoch während des Betriebs von der Maschine demontiert, so werden die Not-Aus Leitungen unterbrochen und die Not-Aus Funktion der Maschine wird aktiviert, sodass die Maschine stoppt.
Wenn ein Handterminal einer Maschine deinstalliert wird, und nicht gleich wieder installiert wird, muss das Gerät weggesperrt werden, um einer Verwechslung mit einem funktionierenden Handterminal zu vermeiden. Die Maschine kann erst wieder in Betrieb genommen werden, wenn ein neues Handterminal installiert wird.
Dies muss in der Betriebsanleitung der Maschine beschrieben sein und obliegt der Verantwortung des Betreibers.

Aus den Gründen, dass das An- und Abstecken nur sehr selten geschieht, und die Maschine im abgesteckten Zustand außer Betrieb ist, wird das Risiko von Unfällen aufgrund nicht betriebsbereiter Not-Aus Schalter als sehr niedrig betrachtet und die Anwendung der rot-gelben Kennzeichnung ist zulässig.

Der rot-gelbe Not-Aus Schalter muss auf jeden Fall in den Not-Aus Kreis der Maschine verschaltet werden, und eine Energieabschaltung für die Maschine bzw. alle zu einer Anlage zusammengeschalteten Maschinen bewirken (Stopp-Kategorien 0 oder 1 nach EN 60204-1).

Funk-Handterminals: Stopp-Schalter

Ein anderer Fall sind drahtlose Handterminals. Diese Geräte sind normalerweise nicht einer bestimmten Maschine zugewiesen, sondern können während des Betriebs der Maschine häufig an- und abgemeldet werden und auch zwischen unterschiedlichen Maschinen gewechselt werden. Dadurch ist der Stopp-Schalter nicht immer betriebsbereit, und der Betriebszustand ist nicht für jedermann eindeutig.
Daher fordert die EN 60204-1:*1997* "Sicherheit von Maschinen - Elektrische Ausrüstung von Maschinen" in Kapitel 9.2.7.3 für Stopp-Funktionen in kabellosen Steuerungen:

"Das Bedienteil, das diese Stopp-Funktion einleitet, darf nicht als Einrichtung zum Stillsetzen im Notfall markiert oder beschriftet sein, auch dann nicht, wenn die Stopp-Funktion, die an der Maschine ausgelöst wird, eine Funktion zum Stillsetzen im Notfall sein kann."

Aus diesem Grund verwendet KEBA für Funk-Handterminals einen Stopp-Schalter, der alle mechanischen Merkmale eines normalen Not-Aus Schalters aufweist, jedoch in neutraler, grauer Farbe gekennzeichnet ist.

Im Unterschied zum rot-gelben Not-Aus besteht keine Notwendigkeit die Stopp-Ausgangs-Signale des Funk Empfangsgerätes in den normalen Not-Aus Kreis der Maschine zu schalten. Er kann auch verwendet werden, um einzelne Sicherheitszonen einer Maschine oder Anlage zu stoppen, wobei

KEBA © KEBA 2006

auch Funktionen wie „Sicherer Betriebshalt“, ausgelöst werden können. Das bedeutet, dass die Antriebskomponenten durch aktive, mit Energie versorgte Antriebe sicher im Stillstand gehalten werden (Stopp-Kategorie 2 nach EN 60204-1). Dies kann den Verlust von Referenzierungen verhindern, und ermöglicht einen rascheren Wiederanlauf der Maschinen.

Der geschulte Bediener des Handterminals bzw. der Maschine kennt die zugewiesene Funktion des Schalters, und weiß über den jeweils aktuellen Betriebsmodus des Gerätes sowie der aktuellen Maschinenzuordnung Bescheid. Somit stellt die Farbgebung für ihn keinen Nachteil dar.
Der Sicherheitsgewinn liegt darin, dass im Fall von nicht aktiven oder nicht zugewiesenen Geräten keine Verwechslungsgefahr mit funktionstüchtigen Not-Aus Schaltern für Dritte besteht.

Temporär steckbare Handterminals: Stopp-Schalter

Manche kabelgebundenen Geräte werden mit einem Schnellverschluss (z.B. Bajonett) versehen, der es ermöglicht, das Handterminal während des Betriebs der Maschine rasch und ergonomisch an- und abzustecken. Solche Handterminals sind für einen häufigen Betriebswechsel zwischen einer oder mehreren Maschinen vorgesehen und werden dort nur temporär, für Inbetriebnahme oder Einrichtarbeiten benötigt.

Mit verschiedenen Maßnahmen, wie z.B. Überbrückungssteckern oder Relaisschaltungen, werden die Not-Aus-Kreise der Maschine bei abgestecktem Handterminal überbrückt, sodass die Maschine im Normalbetrieb auch ohne Handterminal weiterlaufen kann. Start und Bedienung der Maschine können dann über eine unabhängige Bedieneinrichtung erfolgen.

Diese Handterminals teilen die Problematik der Funk-Handterminals, da durch das häufige An- und Abstecken nicht ausgeschlossen werden, dass abgesteckte Handterminals mit wirkungslosen Stopp-Schaltern zwischenzeitlich in einer Maschinenhalle oder Fabrik in der Nähe der arbeitenden Maschinen herumliegen oder -hängen und in Not-Situationen für funktionstüchtige Geräte gehalten werden.

Dieser spezielle Fall wird in den einschlägigen Normen noch eher stiefmütterlich behandelt. In den Produktnormen für Pressen (EN 692:2004, EN 693:2001, EN 12622:2001, EN 13736:2003) findet man aber z.B. den eindeutigen Satz:

„Steckbare Steuerpulte, die entfernt werden können, dürfen keinen Not-Aus-Schalter beinhalten, wenn die Presse bei entferntem steckbaren Steuerpult betrieben werden kann.“

Weitere Produktnormen befinden sich derzeit in Überarbeitung.

In mehreren Gespräche mit externen Prüfstellen und technischen Komitees wurde ebenfalls klar festgestellt, dass temporär steckbare Handterminals, wie Funk-Handterminals zu behandeln sind.
Die Vorgangsweise, die konstruktive Beseitigung der Gefahr durch eindeutige Farbgebung, eventuellen organisatorischen Maßnahmen vorzuziehen,

© KEBA 2006

entspricht auch den bereits oben beschriebenen „Grundsätzen für die Integration der Sicherheit“ der Maschinenrichtlinie und ist daher gesetzlich verpflichtend.

Aus diesem Grund dürfen diese temporär steckbaren Handterminals ebenfalls nur mit einem grauen Stopp-Taster ausgestattet werden.
Da bereits von mehreren Herstellern Handterminals mit grauem Stopp-Schalter auf dem Markt sind, und von benannten Stellen zertifiziert wurden, ist auch der Stand der Technik gegeben.

Hinweise zur Zustimmungseinrichtung

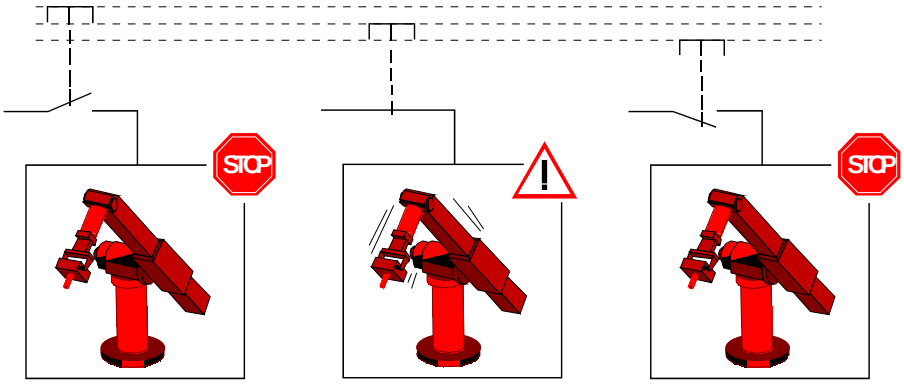
Viele Maschinen verfügen über die Betriebsarten Normalbetrieb und Sonderbetrieb.

Im Normal- (Automatik-) Betrieb erfüllt die Maschine ihren Einsatzauftrag. Die Sicherheit wird in dieser Betriebsart über geschlossene, trennende Schutzeinrichtungen und/oder mit funktionstüchtigen, nicht trennenden Schutzeinrichtungen gewährleistet.

Die Sonderbetriebsarten einer Maschine dienen dazu, den Normalbetrieb aufrecht zu erhalten. Dabei muss die Sicherheit auf eine andere Art als im Normalbetrieb gewährleistet werden, da nun Gefahrenbereiche der Maschine betreten werden können, und gezielte Bewegungen möglich sein müssen. Hier spielt die Zustimmungseinrichtung eine tragende Rolle.

Voraussetzung für die Anwendung ist, dass mittels der Zustimmungseinrichtung, die zu kontrollierende Gefährdung rechtzeitig ausgeschaltet werden, bevor ein Personenschaden auftritt. Hierfür sind eventuell zusätzliche Sicherheitsmaßnahmen, wie eine sicher reduzierte Geschwindigkeit von Antrieben notwendig.

Im Gegensatz zum Not-Aus (bzw. Stopp-) Schalter, befindet sich eine Maschine in den Sonderbetriebsarten mit Zustimmtaster immer im sicheren Zustand, solange dieser nicht betätigt ist (siehe Abbildung).



"Ruhestellung" nicht betätigt "Zustimmung" gedrückt "Panik" stark gedrückt

Zustimmungsfunktionen

KEBA © KEBA 2006

Die Zustimmungs-Funktion ist auch in der EN 60204-1:1997 beschrieben, und entspricht dem Stand der Technik.
Die Panikstellung bei 3-stufigen Zustimmungstastern wurde deshalb eingeführt, da Menschen in Schrecksituationen häufig mit einer reflexartigen Verkrampfung der Gliedmaßen reagieren, und dann die Zustimmungseinrichtung nicht loslassen können. Deshalb führt das feste Durchdrücken des Zustimmungstasters ebenfalls zur Ausschaltung.

Die Steuerung muss so ausgelegt werden, dass die Bewegung nicht direkt mit der Betätigung der Zustimmungseinrichtung eingeleitet wird, sondern erst durch die zusätzliche Betätigung einer Steuerungstaste. Dies können Folientasten am Handterminal sein, oder auch graphische Software-Tasten am Touchscreen.
Auch ein optional vorhandener Joystick kann zur Auslösung des Bewegungssignals verwendet werden.

Für die Zustimmungsfunktion sind lt. EN 60204 nur die Stopp-Kategorien 0 oder 1 erlaubt, d.h. mit dem Stoppen durch den Zustimmungstaster muss in jedem Fall eine Energieabschaltung der Antriebe verbunden sein.

Um eine missbräuchliche dauerhafte Betätigung des Zustimmungstasters mittels mechanischer Fixiereinrichtungen zu verhindern, wird empfohlen, die maximale Dauer einer akzeptierten Zustimmung zu begrenzen. Dies muss durch eine dem Handterminal nachgeschaltete Steuerung realisiert werden.

Eine Zustimmungseinrichtung ist kein Ersatz für eine Zweihandschaltung, wie sie für manche Maschinen (z.B. Pressen) vorgeschrieben wird, und darf daher nicht damit verwechselt werden!
Im Gegensatz zur Zustimmungseinrichtung, zwingt die Zweihandschaltung während der Maschinenbewegung die gefährdeten Gliedmaßen des Bedienpersonals in eine sichere Position.

© KEBA 2006

Elektromagnetische Verträglichkeit

Die Europäische Union verpflichtet die Mitgliedstaaten durch die Richtlinie 89/336/EWG (wird ab 20.7.2007 durch die neue Richtlinie 2004/108/EG ersetzt) ihre Rechtsvorschriften über die elektromagnetische Verträglichkeit anzugleichen. Im folgenden Text wird diese Richtlinie kurz EMV-Richtlinie bezeichnet.
In Europa müssen daher alle in Verkehr gebrachte elektrische und elektronische Betriebsmitteln den grundlegenden Sicherheitsanforderungen der EMV-Richtlinie entsprechen. Gemäß der Richtlinie ist die elektromagnetische Verträglichkeit die Fähigkeit eines Apparates, einer Anlage oder eines Systems, in der elektromagnetischen Umwelt zufriedenstellend zu arbeiten, ohne dabei selbst elektromagnetische Störungen zu verursachen, die für alle in dieser Umwelt vorhandenen Apparate, Anlagen oder Systeme unannehmbar wären.
Neben der oben beschriebenen gesetzlichen Anforderungen ist die zuverlässige Funktion eines elektrischen Betriebsmittel auch ein wesentliches Qualitätsmerkmal eines solchen Betriebsmittels.
Neben grundlegende Informationen zur elektromagnetischen Verträglichkeit beschreiben die nachfolgenden Seiten die Umsetzung der Anforderungen der EMV bei den Handbediengeräten der Produktlinie KeTop.

Elektromagnetische Umwelt – Störquellen, Störsenken und Koppelwege

In der elektromagnetischen Umwelt sind eine Vielzahl von künstlichen aber auch natürlichen Störquellen vorhanden, die die elektrischen und elektronischen Betriebsmittel beeinflussen können.
Das bekannteste natürliche Störphänomen ist die atmosphärische Entladung (Blitzentladung).
Künstliche Störquellen sind einerseits beabsichtigt, wie Mobilfunk, Amateurfunk, TV- und Radiosender andererseits unbeabsichtigt, wie Mikrowellenöfen, Lichtbogenschweißen, Zündsysteme, Hochspannungsanlagen, Elektromotore und elektronische Geräte oder auch elektrostatische Entladungen.

Atmosphärische Entladungen (Blitze):

Atmosphärische Entladungen können direkt oder indirekt auf Versorgungs- bzw. Kommunikationsleitungen erfolgen. Die Folge solcher Einschläge ist bei unzureichendem Schutz eine Zerstörung der elektrischen Betriebsmitteln.
Atmosphärische Entladungen werden entsprechend der internationalen Prüfnorm IEC 61000-4-5 (für Europa EN 61000-4-5) – *Prüfung der Störfestigkeit gegen Stoßspannungen* – simuliert.
Die zutreffende internationale Produktnorm für Steuerungssysteme IEC61131-2 (für Europa EN61131-2) teilt die Einsatzbereiche in Zonen ein. Je nach zutreffender Zone sind höhere oder niedrigere Störpegel zu erwarten. Alle Kemro-K2 Steuerungssysteme und auch die Produkte der Linie KeTop sind für den Einsatz in der Zone B geeignet.

KEBA © KEBA 2006

Die Produktnorm fordert die Prüfung der Störfestigkeit gegen Stoßspannungen nach der IEC 61000-4-5.

Elektrostatische Entladungen (ESD)

Durch Berühren und anschließendes Trennen von Materialien können diese aufgeladen werden. Der Effekt tritt nur dann auf, wenn zumindest eines der beiden Materialien ein Nichtleiter ist. In Folge kann es zu einer raschen Entladung kommen, wenn ein aufgeladener oder durch die Influenz eines elektrostatischen Feldes veränderter Leiter in die Nähe eines metallischen Objektes kommt.
Die nach der Aufladung möglichen Spannungen zwischen den geladenen Körpern erreichen Werte bis über 10 kV.
Die häufigste Erscheinung der elektrostatische Entladung erfolgt zwischen Personen und metallischen Körpern. Da man Entladungen unter 3500V praktisch nicht wahrnimmt und elektronische Bauteile aber bereits bei kleineren Spannungen zerstört werden, passieren ESD-Schädigungen von elektronischen Bauteilen häufig auch unbemerkt.
Zur meßtechnischen Simulation der elektrostatischen Entladung wird der internationale Standard IEC 61000-4-2 (für Europa EN 61000-4-2) verwendet. Die internationale Produktnorm IEC 61131-2 (in Europa EN 61131-2) für speicherprogrammierbare Steuerungen fordert die Prüfung nach der IEC 61000-4-2 und legt dazu die Prüfschärfegrade fest.

Technische Systeme als Störquellen

Technische Systeme können als Störquellen auftreten. Dabei können die Störungen beabsichtigt oder unbeabsichtigt sein. Häufig wird zur Materialbearbeitung auch elektromagnetische Energie verwendet.

Periodisch auftretende Störungen:

- Zündimpulse von Verbrennungsmotoren
- Bürstenfeuer von Kommutatormotoren
- Elektromagnetische Felder von Induktionsöfen, Lichtbogenscheißgeräten, Mikrowellengeräten....
- Pulsströme von Frequenzumrichtern und Schaltnetzteilen
- Elektromagnetische Felder von Radio- und Telekomunikationseinrichtungen

Zufällig auftretende Störungen:

- Zündimpulse von Leuchtstofflampen
- Schaltvorgänge bei induktiven Stromkreisen
- Kontaktprellen beim Schließen oder Öffnen von Schaltkontakten
- Spannungsschwankungen bei Schaltvorgängen mit großen Lasten

Für die oben aufgelisteten beabsichtigten und unbeabsichtigten von technischen Systemen verursachten Störquellen gibt es eine Reihe von Prüfnormen, die diese Störungen simulieren:

© KEBA 2006

- IEC 61000-4-3 Prüfung der Störfestigkeit gegen hochfrequente elektromagnetische Felder
- IEC 61000-4-4 Prüfung der Störfestigkeit gegen schnelle transiente Störgrößen/Burst
- IEC 61000-4-6 Prüfung der Störfestigkeit gegen leitungsgeführte Störgrößen, induziert durch hochfrequente Felder
- IEC 61000-4-8 Prüfung der Störfestigkeit geg. Magnetfelder mit energietechn. Frequenzen
- IEC 61000-4-11 Prüfung d. Störfestigkeit gegen Spannungseinbrüche, Kurzzeit-unterbrechungen und Spannungsschwankungen

Alle hier aufgelisteten internationale Normen gibt es auch als europäische Normen. Die Produktnorm IEC 61131-2 fordert Prüfungen nach diesen Normen und legt dazu die Schärfegrade fest.

Technische Systeme als Störsenken

EMV-Probleme treten erst bei Funktionsstörungen an den Störsenken auf. Je nach dem Grad der Störfestigkeit gegen elektromagnetische Einflüsse identifiziert man folgende Störsenken:

Steuerungssysteme sind ohne integrierte Schaltkreise undenkbar und wären daher auch ohne geeignete EMV-Maßnahmen nicht ausreichend störfest.

Koppelwege

Die Übertragung von Störsignalen einer Störquelle zu einer Störsenke kann über verschiedene Koppelwege erfolgen.
Sehr häufig bestehen Koppelwege aus zwei oder mehreren parallel eng nebeneinander verlaufenden Leitungen. Die Kopplung ist eine Feldkopplung, die bei niedrigen Frequenzen entweder über das elektrische Feld (kapazitive Kopplung) oder über das magnetische Feld (induktive Kopplung) erfolgt.

KEBA © KEBA 2006

Bei hohen Frequenzen und entsprechender Ausdehnung der parallel verlaufenden Leitungen spricht man wegen der engen Verknüpfung der beiden Feldtypen von einer elektromagnetischen Kopplung.
Galvanische Kopplung kann entstehen, wenn Stromkreise der Störquelle und Stromkreise der Störsenke gemeinsame Leitungsteile haben.
Besteht zwischen Störquelle und Störsenke ein großer räumlicher Abstand, dann spricht man von Strahlungskopplung.

EMV-Maßnahmen

Grundsätzlich sollten alle Geräte bereits so konstruiert sein, dass diese in der vorgesehenen Umgebung zuverlässig funktionieren und dabei auch nicht andere Systeme stören. Alle Produkte der Kemro Produktlinie (K2 Steuerungssysteme und KeTop) erfüllen diese Anforderungen und es müssen keine der nachfolgend beschriebenen EMV-Maßnahmen angewandt werden, wenn die vorgeschriebenen Zusatzgeräte, Kabeln und Verdrahtung verwendet werden.
Es können aus verschiedenen Gründen jedoch zusätzliche EMV-Maßnahmen notwendig sein.
Der nachfolgende Text soll dem Anwender helfen, eventuell zusätzliche EMV-Maßnahmen korrekt umzusetzen.

Schirmung

Häufig treten bei Produkten Störaussendungsprobleme und Störfestigkeitsprobleme gemeinsam auf. Es wirken auch EMV-Maßnahmen meistens gleichermaßen bei Störaussendungsproblemen und bei Störfestigkeitsproblemen.
Schirmung erfüllt grundsätzlich zwei Aufgaben. Einerseits wird das Eindringen von elektromagnetischen Feldern in empflindliche elektronische Schaltungsteile verhindert und andererseits soll auch die Abstrahlung von elektromagnetischen Feldern verhindert werden.
Ein vollständiger EMV-Schirm besteht aus einem Schirmgehäuse, welches die empfindliche Elektronik schützt bzw. deren Störaussendung verhindert und aus Kabelschirmen die die empfindliche Schnittstellensignale schirmen bzw. die Störaussendung der Geräte über deren Schnittstellen verhindern.

Der Kabelschirm verbindet grundsätzlich zwei Schirmgehäuse miteinander und muss daher an beiden Enden (Anschlussstellen) direkt mit den Schirmgehäusen der empfindlichen Elektronik verbunden werden.
Besonders zu beachten ist die Verbindung zwischen Kabelschirm und den Schirmgehäusen der Geräte. Damit die Schirmwirkung auch bei höheren Frequenzen erhalten bleibt, muss darauf geachtet werden, dass der Anschluss des Kabelschirmes möglichst räumlich erfolgt und damit als räumliche Fortsetzung des Geräteschirmgehäuses wirken kann. Der Anschluss über einen einzelnen Draht (Pig tail) ist dafür nicht geeignet.

Erdung von Kabelschirmen und Schirmgehäusen kann aus sicherheitstechnischen Gründen notwendig sein, ist aber keine wirkungsvolle EMV-Maßnahme.

© KEBA 2006

Entstörfilter, Filterbauelemente

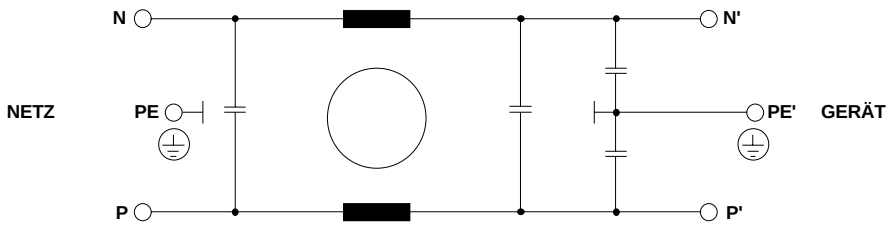
Filterung ist immer dann notwendig, wenn ungeschirmte Signal- und Stromversorgungsleitungen in geschirmte Bereiche geführt werden. Häufig führen diese Leitungen neben dem Nutzsignal auch Störsignale, die nicht in die geschirmten Bereiche gelangen dürfen. Filter sollen daher die Störfestigkeit der Geräte sicherstellen, aber auch die Störaussendung der Geräte über die ungeschirmten Leitungen verhindern.
Ungeschirmte Leitungen werden normalerweise dann verwendet, wenn die geführten Nutzsignale sehr niederfrequent sind. Durch frequenzselektive Filterung mit Hilfe von Tiefpassfiltern erreicht man eine Trennung der üblicherweise höherfrequenten Störsignale von den Nutzsignalen.
Tiefpassfilter müssen so dimensioniert werden, dass die niederfrequenteren Nutzsignale ungehindert passieren können und die höherfrequenteren Störsignale gefiltert werden.
Häufig sind zur Filterung mehrstufige Filter notwendig. Fast alle Filterkombinationen beinhalten Y-Kondensatoren, das sind Kondensatoren die zur Ableitung von Störströmen mit dem Filtergehäuse verbunden sind. Für die richtige Funktion dieser Filter muss das Gehäuse mit einem stabilen Bezugspotenzial verbunden sein.

Da die Störsignale nicht in den geschirmten Bereich gelangen sollen, muss das Potenzial des Schirmgehäuses auch das Bezugspotenzial für die Filterschaltungen sein.

Filterschaltungen oder Filterbauelemente müssen daher genau beim Eintritt der Leitungen in den geschirmten Bereich platziert werden. Werden die Filter innerhalb oder außerhalb der Schirmwand platziert, kann es zur Feldkopplung zwischen den gefilterten und den ungefilterten Leitungen kommen.
Der Anschluss an das Bezugspotenzial muss möglichst niederimpedant erfolgen und daher ist eine großflächige Kontaktierung der normalerweise metallischen Filtergehäuse mit der Schirmwand des Gerätes notwendig. Für eine solche Kontaktierung dürfen keine einzelne Drahtstücke (Pig tails) verwendet werden.

Netzfilter

Typische Netzfilter haben ein Metallgehäuse, das aus sicherheitstechnischen Gründen mit dem Schutzleiter (PE) verbunden werden muss und bestehen (siehe Bild) aus einer stromkompensierten Drossel, aus Y-Kondensatoren, die auf einer Seite mit dem Metallgehäuse verbunden sind und aus X-Kondensatoren (Anschlüsse zwischen den Phasen bzw. zwischen Phase und Nullleiter).



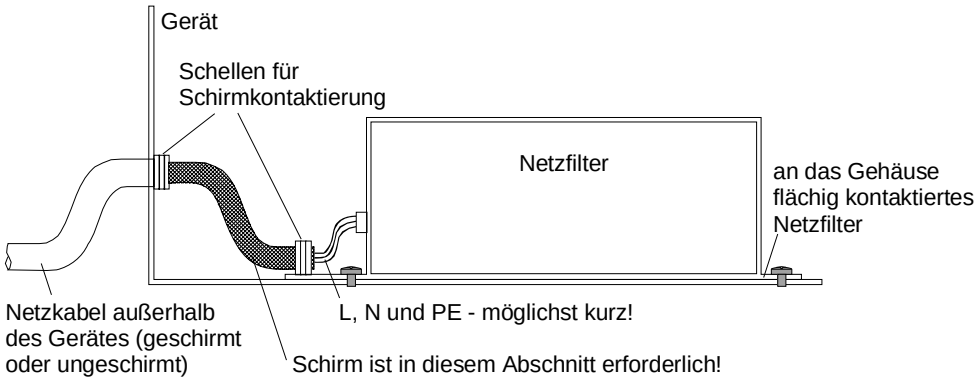
Netzfilter

KEBA © KEBA 2006

Die gewünschte Filterwirkung wird wesentlich von der richtigen Kontaktierung des Filtergehäuses mit dem Schirmgehäuse des Gerätes beeinflusst. Die Netzfilter müssen daher direkt beim Eintritt in das Schirmgehäuses des Gerätes platziert werden und das Metallgehäuse des Filters muss großflächig mit dem Schirmgehäuse des Gerätes verbunden werden.

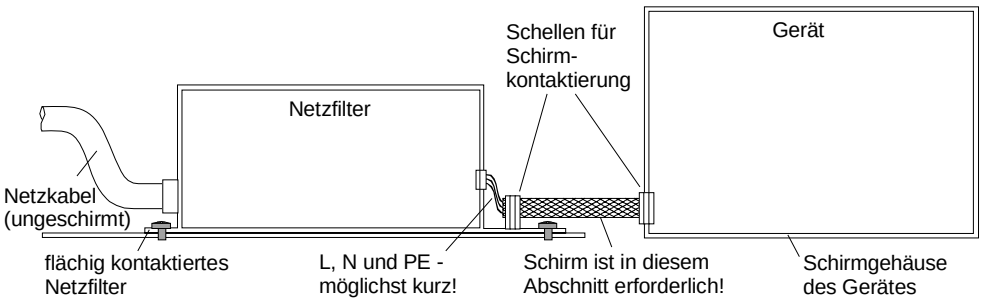
Ist die Platzierung beim Eintritt aus Platzgünden nicht möglich, dann müssen teilweise geschirmte Anschlussleitungen verwendet werden. Dabei ist folgendermaßen vorzugehen:

Wird das Filter innerhalb des Schirmgehäuses des Gerätes platziert, dann muss das Netzanschlusskabel vom Eintritt beginnend bis zum Netzfilter geschirmt werden und der Kabelschirm beidseitig kontakiert werden. Dadurch wird verhindert, dass sich Störungen auf der Netzleitung innerhalb des Gerätes ausbreiten können. Es wird aber auch verhindert, dass eventuell innerhalb des Gerätes vorhandene elektromagnetische Felder auf die Netzleitung einkoppeln.



Richtiger Anschluss eines Kompaktnetzfilters

Wird das Netzfilter außerhalb des Schirmgehäuses platziert, dann muss die Geräteanschlussleitung beginnend vom Netzfilter bis zum Schirmgehäuse des Gerätes geschirmt werden und der Kabelschirm muss wieder beidseitig kontaktiert werden. Dadurch wird verhindert, dass Störungen, die vom Gerät erzeugt werden, nach außen übertragen werden und dort auf andere Systeme einkoppeln können. Außerdem wird verhindert, dass in die bereits gefilterten Anschlussdrähte wiederum Störsignale eingekoppelt werden.



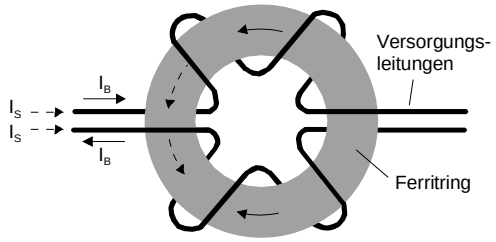
Richtiger Anschluss eines Kompaktnetzfilters

© KEBA 2006

Bei der Auswahl von Netzfiltern ist zuerst darauf zu achten, welche Frequenzbereiche gefiltert werden. Sollen auch Frequenzen im MHz-Bereich gefiltert werden, dann soll ein solches Netzfilter ein Metallgehäuse haben. Netzfilter mit Kunststoffgehäusen sind normalerweise dafür nicht geeignet.

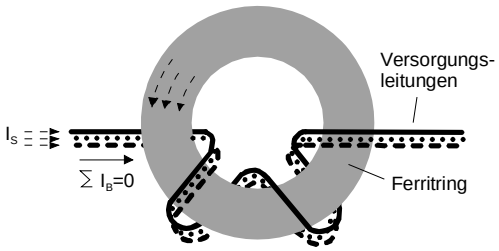
Stromkompensierende Drosseln

Manchmal schreiben Hersteller von Geräten die Anwendung von stromkompensierenden Drosseln oder den Einbau von Ferritringen in die Stromversorgungsleitung vor. Beispiele dafür findet man immer wieder bei Motorleitungen, die von Frequenzumrichtern gespeist werden.
Solche Drosseln (siehe folgende Abbildungen) unterdrücken Gleichtaktstörströme (I_S) und lassen die Betriebsströme (I_B) ungehindert passieren. Es ist darauf zu achten, dass die Betriebsströme im Kern sich aufhebende magnetische Felder erzeugen, damit die Ferritkerne durch den Betriebsstrom nicht gesättigt werden. Die Platzierung solcher Drosseln muss auch direkt beim Eintritt in das Schirmgehäuse des Gerätes erfolgen.



I_S ... Störstrom
I_B ... Betriebsstrom

Stromverlauf bei einer stromkompensierenden Drossel



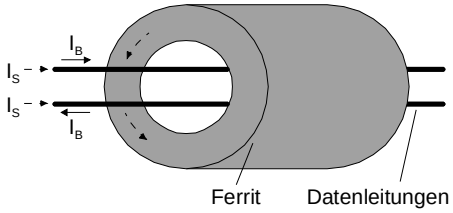
I_S ... Störstrom
I_B ... Betriebsstrom

Stromverlauf bei einer stromkompensierenden Drossel

Ferritte

Ferritte über Datenleitungen wirken ähnlich wie stromkompensierende Drosseln. Die Nutzsignale (siehe Abbildung) können ungehindert passieren und die Störsignale werden gedämpft. Ein wesentlicher Unterschied zu den stromkompensierenden Drosseln ist, dass bei den Ferritten der Einsatzbereich bei höheren Frequenzen erfolgt und daher die Materialverluste für die Dämpfung verantwortlich sind und die Störung nicht wie bei Induktivitäten reflektiert, sondern in Wärme umgewandelt wird.

KEBA © KEBA 2006



I_S ... Störstrom
I_B ... Betriebsstrom

Stromverlauf bei Ferriten

Vermeidung von Kopplungen

Durch getrennte Verlegung von Kabeln kann die Kopplung zwischen störempfindlichen Signalen (Bsp.: analoge Signale) und solchen Leitungen die Störsignale führen (Bsp.: Motorleitungen) verringert werden.

EMV Maßnahmen bei KeTop

Das KeTop ist für industrielle Anwendungen bestimmt. In diesem Umfeld treten zum Teil erhebliche elektromagnetische und elektrostatische Störungen auf. Aus diesem Grund wurde bereits bei der Konstruktion des Geräts besonderer Wert auf Störunempfindlichkeit und Datensicherheit gelegt. Das umgesetzte und nachfolgend beschriebene EMV-Konzept berücksichtigt alle oben genannten Störquellen und die möglichen Koppelwege.

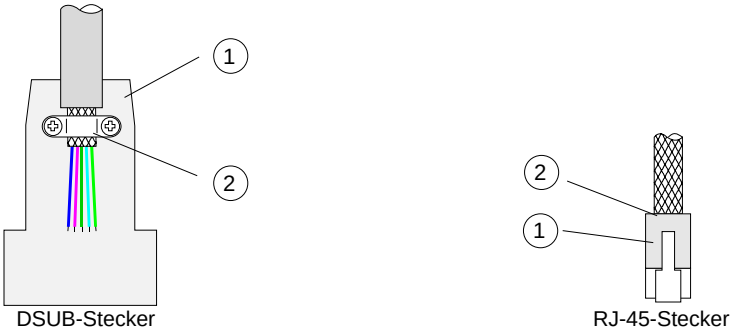
- Alle Schirm- und Filtermassnahmen (gefilterte Versorgungsspannung) sind im KeTop direkt auf der Flachbaugruppe realisiert.
- Das KeTop-Kabel gewährleistet durch speziellen Kabelaufbau die Störfestigkeit auch bei größeren Anschlusslängen.
  Dh. die Datenleitungen (Kommunikationssignale) sind geschirmt und werden von den Steuerleitungen (Versorgung, Zustimmungstaster, Not-Aus, Schlüsselschalter usw.) getrennt im KeTop-Kabel geführt.
- Üblicherweise werden Versorgungsleitungen im Schaltschrank ungeschirmt geführt. Sie werden daher außerhalb des KeTop-Kabelschirms geführt, um eine Kopplung mit den empfindlichen Datenleitungen zu vermeiden.
- Die ungeschirmten Steuer- und Versorgungsleitungen sind entweder beim Eintritt in das Schirmgehäuse des KeTop gefiltert oder von der empfindlichen Elektronik so getrennt, dass eventuell von diesen Leitungen geführte Störsignale die interne Elektronik des KeTops nicht beeinflussen können.
- Es ist keine getrennte Verlegung des KeTop-Kabels notwendig.

© KEBA 2006

Schirmverbindungen

Der Kabelschirm des KeTop-Kabels kann als Erweiterung des KeTop-Schirmgehäuses (=Flachbaugruppe) bis zum Schirmgehäuse des Kommunikationspartners (zB: SPS) betrachtet werden. Daraus läßt sich ableiten, daß die Schirmverbindungen des Kabelschirmes zu den Geräteschirmen einen wesentlichen Beitrag zur Störunempfindlichkeit des KeTops liefert. Verbindungen zu Erdpotential sind für Schirmanschlüsse nicht notwendig. Die Schirmverbindung im KeTop erfolgt über den RJ-45-Stecker im Anschlussschacht.
Der Anschluss des KeTops über die Anschlussbox (zB CB211) garantiert eine sichere Schirmverbindung bis zum Kommunikationspartner. Es wird daher dringend empfohlen, eine passende KEBA-Anschlussbox zu verwenden.
Alle bei KEBA erhältlichen Verbindungskabel (KeTop TTxxx, KeTop ICxxx, KeTop XD040, KeTop CD040) gewährleisten eine ordnungsgemäße Schirmverbindung.
Bei selbstkonfektionierte Kabeln müssen folgende Bedingungen für die Schirmung der Datenleitungen erfüllt sein:

- Der Kabelschirm muss bei jeder Steckerausführung (DSUB oder RJ-45) großflächig am Steckergehäuse aufliegen.

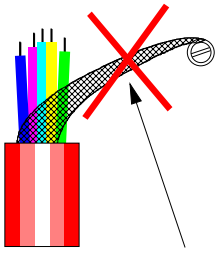


DSUB-Stecker RJ-45-Stecker

1..... Metallisiertes oder metallisches Gehäuse
2..... Schirm großflächig anschließen

Ordnungsgemäßer Schirmanschluss bei DSUB und RJ-45-Stecker

- Die Verwendung von Pig Tails (Kabelschwänzchen) zur Kontaktierung des Schirms über Stiftkontakte ist ungeeignet. Die Induktivität solcher Pig Tails stellt für höherfrequente Störungen einen hochohmigen Widerstand dar, dies bedeutet eine scheinbare Unterbrechung des Kabelschirms. Störungen werden somit nicht mehr abgeleitet und wirken damit direkt auf die innenliegenden Leitungen.



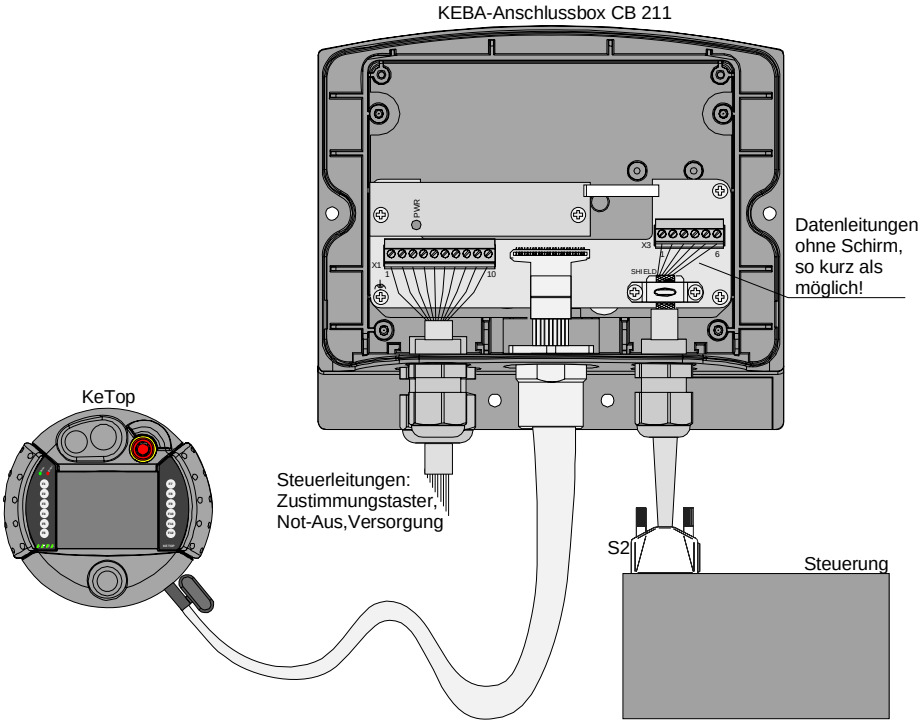
keine „Kabelschwänzchen“

Unzureichende Kontaktierung eines Kabelschirms

KEBA © KEBA 2006

Schirmung innerhalb des Schaltschrankes

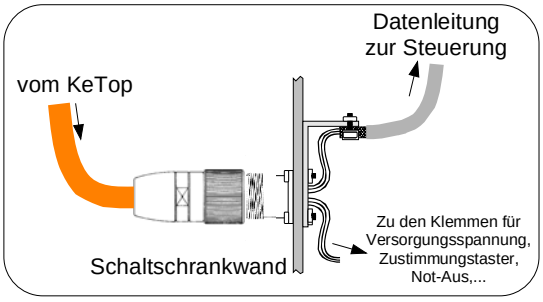
In vielen Fällen sind im Schaltschrank eine Reihe von Störquellen wie zB Servoantriebsmodule, Transformatoren, Schütze und Relais vorhanden. Es ist deswegen notwendig, den Kabelschirm vom Steckergehäuse (Schaltschrank) bis zur Steuerung weiterzuführen (durchgängige Verbindung vom Handterminal bis zur Steuerung).
Bei Verwendung der passenden Anschlussbox und eines geschirmten Kabels für die Datenleitung von Anschlussbox bis zur Steuerung ist die durchgängige hochfrequenztaugliche Verbindung des Schirmes vom KeTop bis zur Steuerung garantiert.



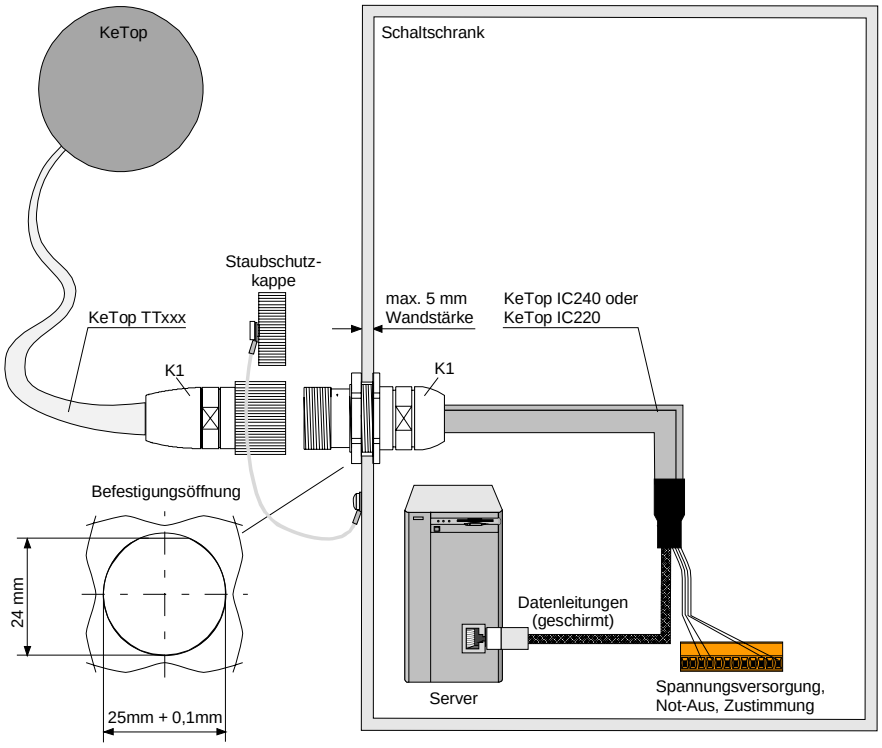
Beispiel eines ordnungsgemäßen Schirmanschlusses in einer KEBA-Anschlussbox

Kann aus irgendwelchen Gründen keine Anschlussbox verwendet werden, dann kann die durchgängige Verbindung Schirmes bei weniger kritischen Schnittstellentypen wie RS232 durch Kontaktierung des Steckergehäuses mit dem Schaltschrank und innerhalb des Schaltschrankes durch Kontaktierung des Schirmes mit dem Schaltschrank (durch Schirmschellen) erfolgen. Je kleiner der Abstand zwischen der Kontaktierung des Steckergehäuses auf dem Schaltschrank zu der Kabelschelle im Schaltschrank ist, desto besser ist die Schirmdämpfung.

© KEBA 2006



Die Störfestigkeit des Gerätes mit der oben dargestellten Anschlussart wird auch entscheidet von der ausreichenden Trennung der Steuersignale von den Signalen der Datenleitung beeinflusst. Je besser die Trennung der beiden Signaltypen gelingt und je kürzer die Schirmverbindung ist, desto höher wird auch die Störfestigkeit des gesamten Systems sein.
Wird als Kommunikationsschnittstelle Ethernet verwendet (nur bei KeTop T100) muss für den Anschluss an ein Steuerungssystem eines der beiden dafür vorgesehenes Anschlusskabel IC020 bzw. IC040 verwendet werden. Beide Kabel führen die Ethernetsignale auf einen geeigneten Stecker (RJ45) und stellen damit eine durchgängige Schirmverbindung und auch den geforderten Wellenwiderstand des Kabels her.



Anschlusskabel IC020 bzw. IC040 im Schaltschrank

KEBA © KEBA 2006

Liste der zutreffenden EG-Richtlinien und angewandten Normen

EG Richtlinien

98/37/EG	Maschinenrichtlinie mit der Änderung 98/79/EG
89/336/EWG	EMV-Richtlinie mit den Änderungen 91/263/EWG und 93/31/EWG

Normen

Zur Überprüfung der Konformität des KeTops mit den Richtlinien wurden die folgenden rechtlich unverbindlichen europäischen Normen angewendet.

Überprüfung der Konformität mit der Maschinenrichtlinie

EN 418:1992	Sicherheit von Maschinen; Not-Aus-Einrichtung, funktionelle Aspekte; Gestaltungsleitsätze
EN 954-1:1996	Sicherheit von Maschinen – Sicherheitsbezogene Teile von Steuerungen - Teil 1: Allgemeine Gestaltungsleitsätze
EN 60204-1:1997, Kap.9, Kap.10	Sicherheit von Maschinen – Elektrische Ausrüstung von Maschinen - Teil 1: Allgemeine Anforderungen

Überprüfung der Konformität mit der EMV-Richtlinie

EN 61131-2:2003 Kap 8, 9	Speicherprogrammierbare Steuerungen - Teil 2: Betriebsmittelanforderungen und Prüfungen

Damit ist auch die Übereinstimmung mit folgende Normen gegeben:

EN 61000-6-2:2001	EMV Fachgrundnorm Störfestigkeit für Industriebereich
EN 61000-6-4:2001	EMV Fachgrundnorm Störaussendung für Industriebereich

Sonstige Normen

Bei der Ausarbeitung des Sicherheitskonzeptes wurden zusätzlich die folgenden rechtlich unverbindlichen europäischen Normen in Teilaspekten zu Rate gezogen:

Allg. Vorgehensweisen und Sicherheitsprinzipien

EN 292-1:1991	Sicherheit von Maschinen – Grundbegriffe, allgemeine Gestaltungsleitsätze - Teil 1: Grundsätzliche Terminologie, Methodik
EN 292-2:1991	Sicherheit von Maschinen – Grundbegriffe, allgemeine Gestaltungsleitsätze - Teil 2: Technische Leitsätze

Ausführung der Zustimmeinrichtung

EN 954-1:1996	Sicherheit von Maschinen – Sicherheitsbezogene Teile von Steuerungen - Teil 1: Allgemeine Gestaltungsleitsätze
EN 60204-1:1997	Sicherheit von Maschinen - Elektrische Ausrüstung von Maschinen - Teil 1: Allgemeine Anforderungen

© KEBA 2006

Ausführung des Stopp-Schalters bzw. Not-Aus Schalters

EN 418:1992	Sicherheit von Maschinen; Not-Aus-Einrichtung, funktionelle Aspekte; Gestaltungsleitsätze
EN 60204-1:1997 Kap. 9, 10	Sicherheit von Maschinen - Elektrische Ausrüstung von Maschinen - Teil 1: Allgemeine Anforderungen

Ergonomie

EN 614-1:1995	Sicherheit von Maschinen - Ergonomische Gestaltungsgrundsätze - Teil 1: Begriffe und allgemeine Leitsätze
EN 894-1:1997	Sicherheit von Maschinen - Ergonomische Anforderungen an die Gestaltung von Anzeigen und Stellteilen - Teil 1: Allgemeine Leitsätze für Benutzer-Interaktion mit Anzeigen und Stellteilen
EN 894-2:1997	Sicherheit von Maschinen - Ergonomische Anforderungen an die Gestaltung von Anzeigen und Stellteilen - Teil 2: Anzeigen
EN 894-3:1997	Sicherheit von Maschinen - Ergonomische Anforderungen an die Gestaltung von Anzeigen und Stellteilen - Teil 3: Stellteile

Festigkeit und Dichtheit des Gehäuses

EN 60529:1991	Schutzarten durch Gehäuse
EN 61131-2:2003 Kap. 12	Speicherprogrammierbare Steuerungen - Teil 2: Anforderungen und Tests

Elektrische Sicherheit u. Brandschutz

EN 61131-2:2003 Kap. 11	Speicherprogrammierbare Steuerungen - Teil 2: Anforderungen und Tests

Anforderungen an Umwelt- und Umgebungsbedingungen

EN 61131-2:2003 Kap. 4	Speicherprogrammierbare Steuerungen - Teil 2: Anforderungen und Tests
IEC 721-2-1	Classification of enviromental conditions

Für den Amerikanischen Markt wurden außerdem folgende Normen berücksichtigt:

UL Prüfung für Industrielle Steuerungseinrichtungen

UL 508, 2001 (=CSA C22.2 No.14)	Industrial Control Equipment

UL Prüfung für Robotik Anwendugen

UL 1740, 1998	Industrial Robots and Robotic Equipment

KEBA © KEBA 2006